HOW TO USE DATA API

1. How to use DATA API via OAuth2 - technical part 2
1.1 Get country identifiers. 3
1.2 Search a company (identification of a company in the database) 3
1.2.1 Search company by company info 3
1.2.2 Search company by one of the externalId 5
1.2.3 Search company by companyId 8
1.3. Request a company identification. 10
1.3.1 Request the identification. 10
1.3.2 Request the needed product. 12
1.3.3 Request all identifications 13
1.4. Ordering a product 14
1.4.1 Ordering a product for an identified company: 14
a. Ordering a Full Report via companyId: 14
b. Ordering a Snapshot report (light report) via externalId: 14
c. Ordering a Sector Risk Assessment 15
d. Ordering Financial products 18
1.4.2 Ordering a product for a company not identified. 23
1.4.3 Ordering a product using Callback mechanism. 24
1.5. Checking order status 25
1.6. Retrieving the data 25
1.7 Request Monitoring 26
1.8 Retrieving content of the report / product ordered via monitoring order 28
1.9 Retrieving monitoring notifications. 29
2. Use cases 32
2.1 Ordering multiple products for same company 32
2.1.1 Ordering 2 assessments in one call 32
2.1.2 Ordering other products for the same company 40
2.2 Ordering Late Payment Index product for a company 45

1. How to use DATA API via OAuth2 - technical part

The main functional flows are represented in https://coface.github.io/DataAPI.html#process

After the proper authentication, always the customer needs to:

- start with Searching and identifying a company,

- check the products availability,

- order a product,

- **check the status of the order** and

- Retrieve the data.

All functional calls below can be found in the Postman collection uploaded under https://coface.github.io/DataAPI.html#documentation

Remark!! In the sample calls which follow, the variable *{{dataAPIBaseUrl}}* = **https://icon-api-test.coface.com/dataapi-v1**

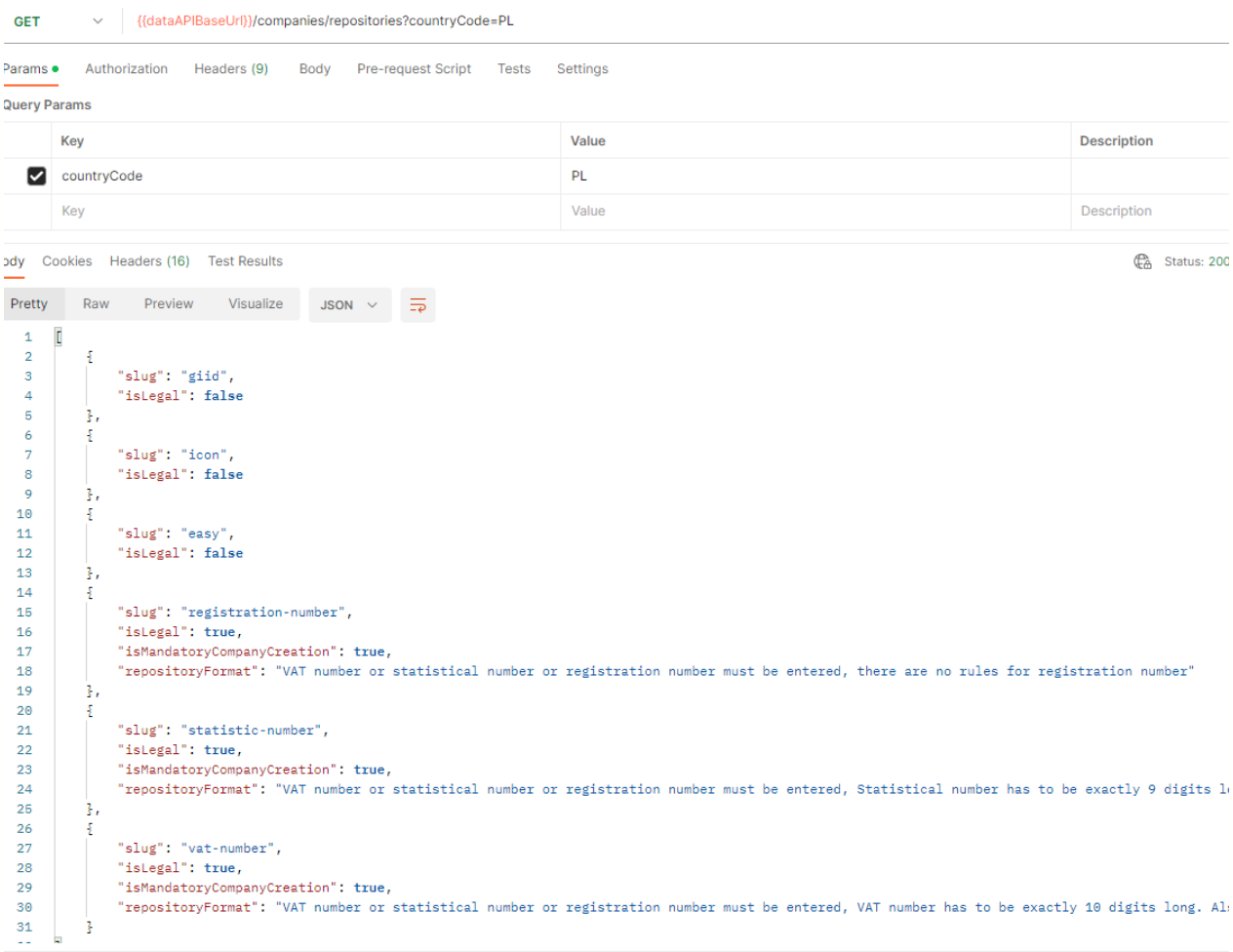
Prerequisite!! In order to perform the following calls, you should already have an user connected via AWS Dev portal. Please check this https://coface.github.io/utilities/iCON_API_UserConfiguration_DeveloperPortal.pdf , in case you do not have yet a connected and authorized user.

1.1 Get country identifiers.

- Allows to obtain details of identifiers in specific countries which can be used for unique company identification (check which identifier is legal per country)
- Request can be done by specifying country ISO Code and in return customer get list of identifiers used in specific country.

Request: ***GET*** *{{dataAPIBaseUrl}}/companies/repositories?countryCode=PL*

Response:



1.2 Search a company (identification of a company in the database)

There are several options for searching and identification of a company:

1.2.1 Search company by company info

Allows to perform search by name and countryCode mostly. Search can be performed using **minimum** companyName (or parts of the name) and countryCode:

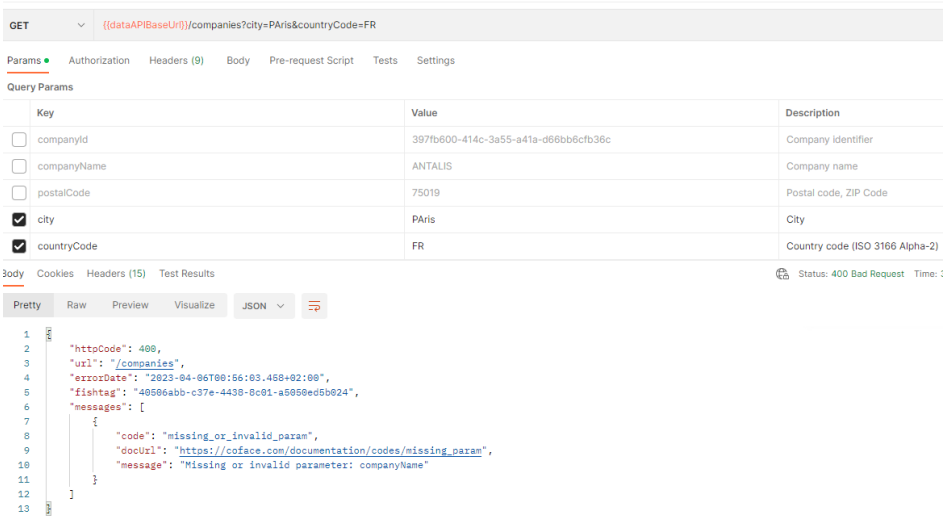
GET *{{dataAPIBaseUrl}}/companies?companyName=ANTALIS&countryCode=FR*=> 28 companies returned

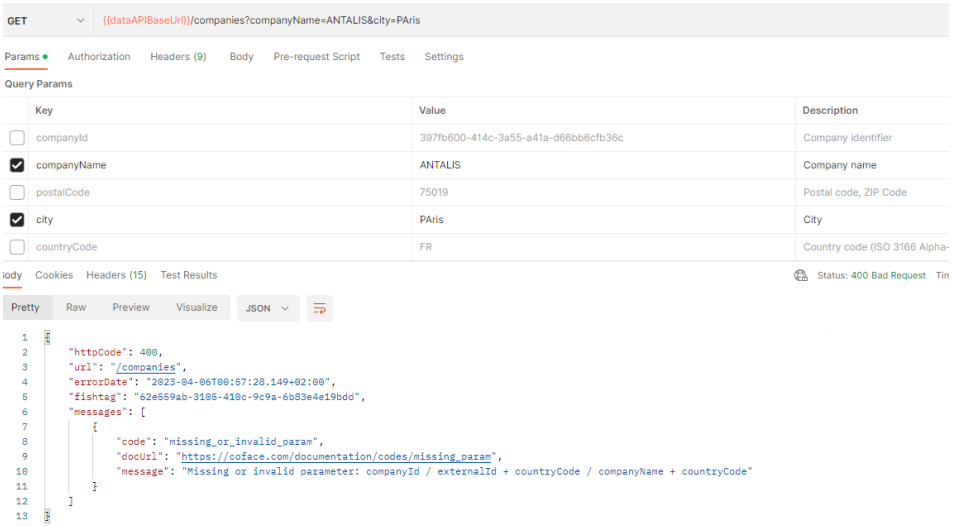
If more information is provided, like postalCode and city, more accurate the search is and company is identified:

GET *{{dataAPIBaseUrl}}/companies?companyName=ANTALIS&city=PAris&countryCode=FR =>* 6 companies returned

GET *{{dataAPIBaseUrl}}/companies?companyName=ANTALIS&postalCode=75019&city=PAris&countryCode=FR* => 1 result returned. Company is identified.

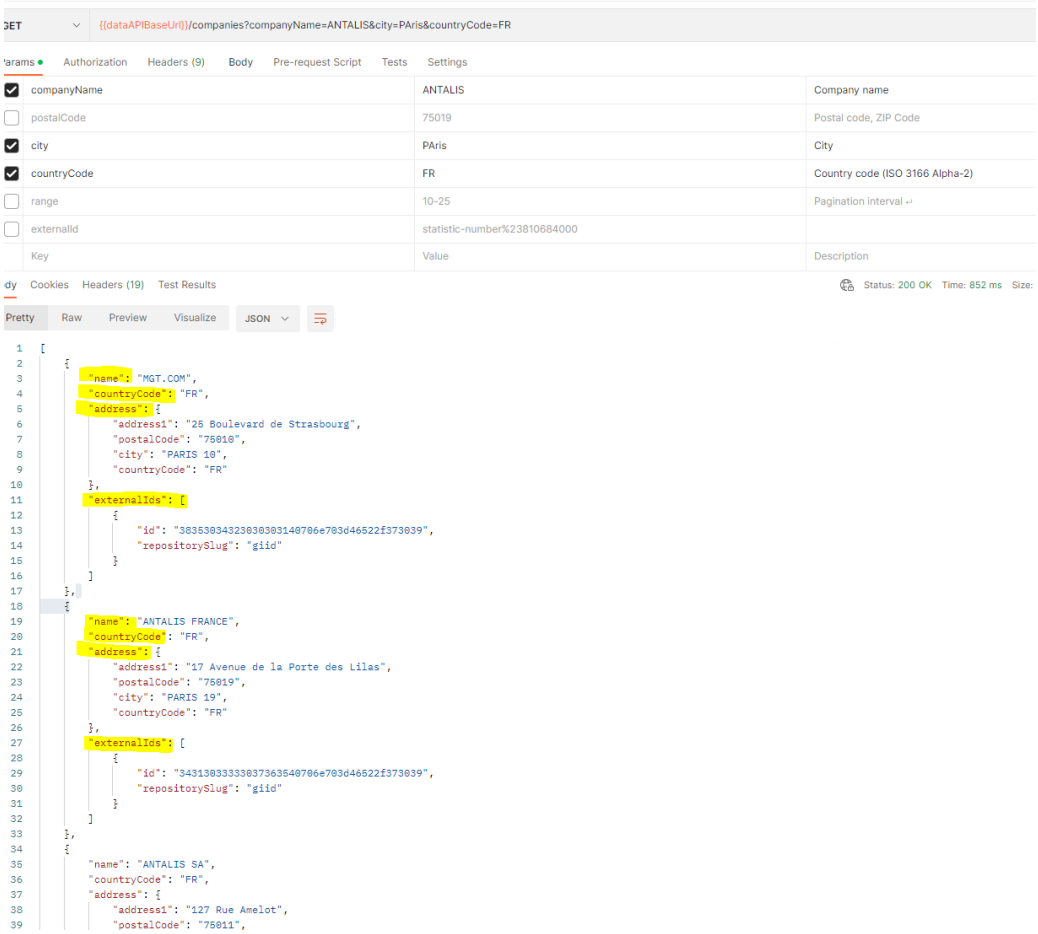
Error case: If combination companyName + countryCode is not provided, 400 Bad request is returned:





Success case – Returns list of companies found with the given criteria. For each company from the list, the response includes: name, countryCode, address and externalIds.

ExternalIds can be also used further for better identification of the company, checking the availability of the products or for ordering products for that corresponding company.



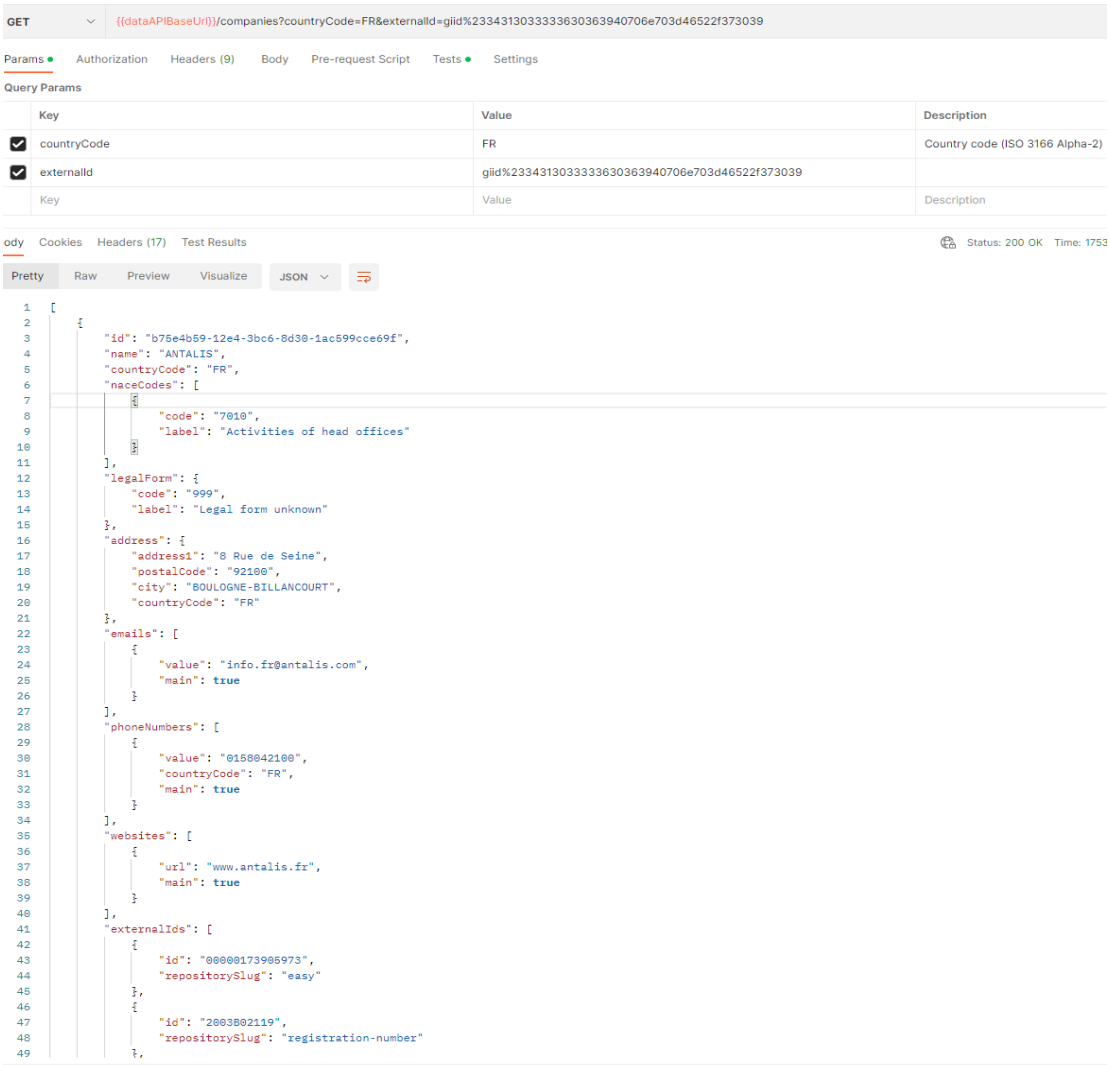
1.2.2 Search company by one of the externalId

Allows to search for a specific company by providing combination of countryCode + externalId. ExternalId can be one of the national identifiers or one of internal Coface IDs. It was either returned by previous method (using companyName + countryCode) either was already known (e.g.VAT number).
Parameter **externalId** will be used with value {repositorySlug}#{id}.

Response returns either 1 unique result or no results.

Identifying company using giid number:

GET {{dataAPIBaseUrl}}/companies?countryCode=FR&externalId=giid%233431303333363036394070 6e703d46522f373039



GET {{dataAPIBaseUrl}}/companies?countryCode=FR&externalId=giid%233431303333363036394070 6e703d46522f373039

Params • Authorization Headers (9) Body Pre-request Script Tests • Settings

Query Params

Key	Value	Description
countryCode	FR	Country code (ISO 3166 Alpha-2)
externalId	giid%233431303333363036394070 6e703d46522f373039	
Key	Value	Description

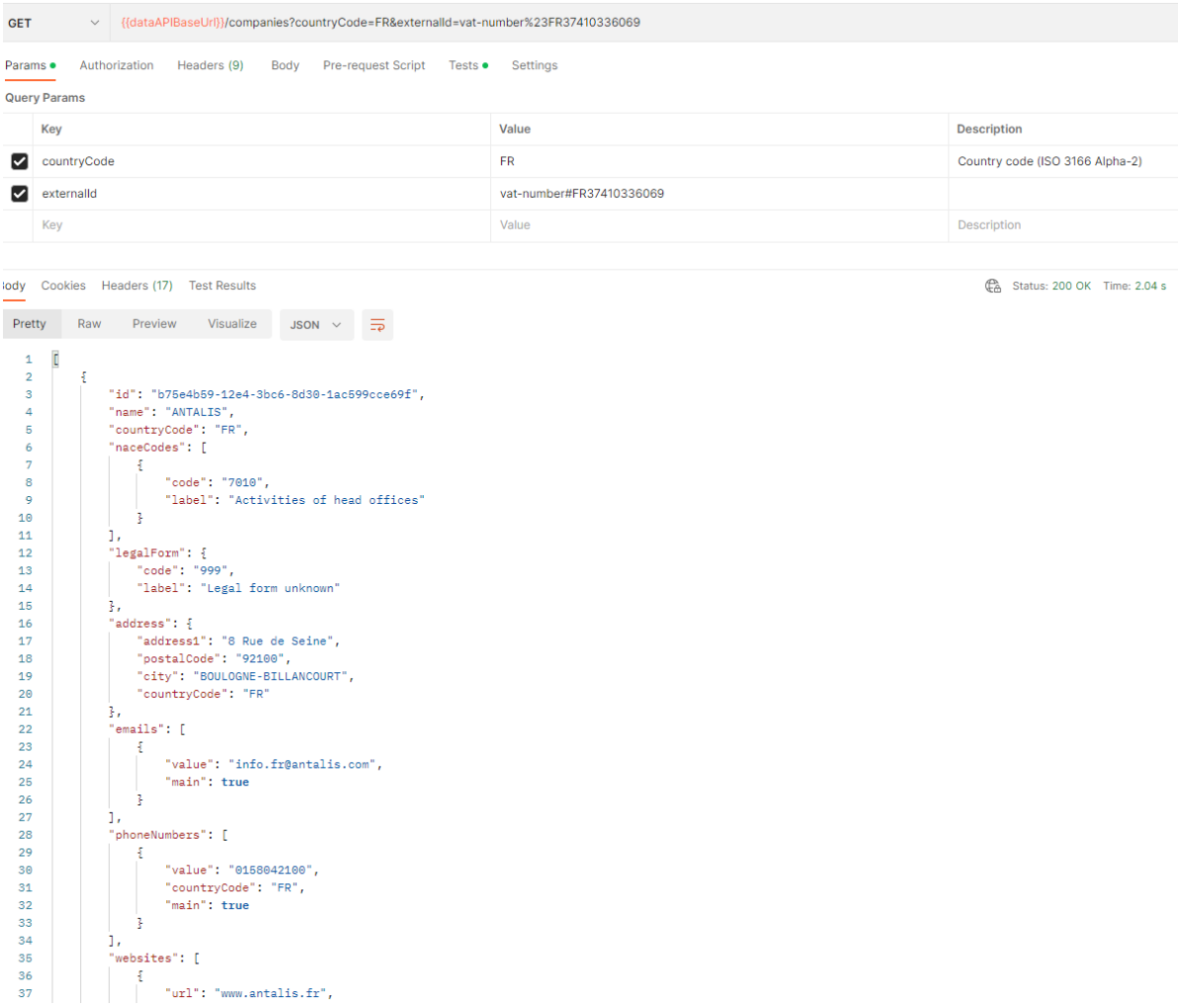
ody Cookies Headers (17) Test Results Status: 200 OK Time: 1753

Pretty Raw Preview Visualize JSON


```
[
    {
        "id": "b75e4b59-12e4-3bc6-8d30-1ac599cce69f",
        "name": "ANTALIS",
        "countryCode": "FR",
        "naceCodes": [
            {
                "code": "7010",
                "label": "Activities of head offices"
            }
        ],
        "legalForm": {
            "code": "999",
            "label": "Legal form unknown"
        },
        "address": {
            "address1": "8 Rue de Seine",
            "postalCode": "92100",
            "city": "BOULOGNE-BILLANCOURT",
            "countryCode": "FR"
        },
        "emails": [
            {
                "value": "info.fr@antalis.com",
                "main": true
            }
        ],
        "phoneNumbers": [
            {
                "value": "0158042100",
                "countryCode": "FR",
                "main": true
            }
        ],
        "websites": [
            {
                "url": "www.antalis.fr",
                "main": true
            }
        ],
        "externalIds": [
            {
                "id": "00000173905973",
                "repositorySlug": "easy"
            },
            {
                "id": "2003B02119",
                "repositorySlug": "registration-number"
            },
```

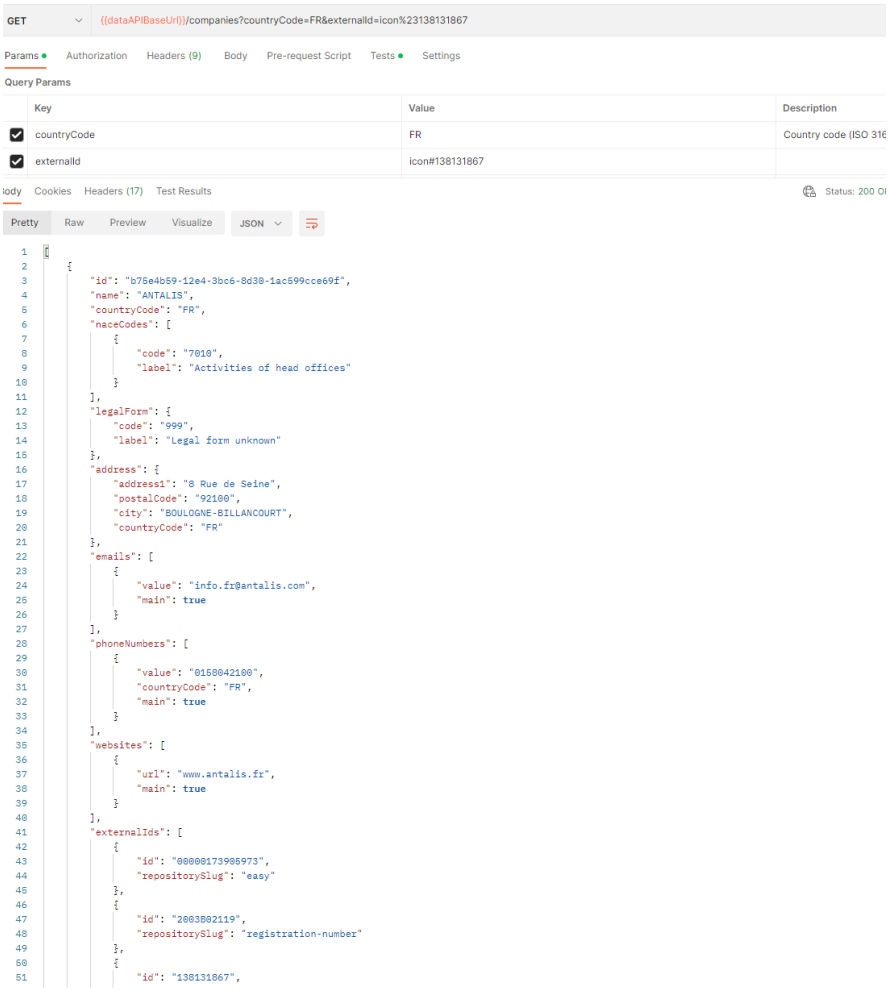
Identifying company using VAT number:

GET {{dataAPIBaseUrl}}/companies?countryCode=FR&externalId=vat-number%23FR37410336069



Identifying company using ICON number:

GET {{dataAPIBaseUrl}}/companies?countryCode=FR&externalId=icon%23138131867

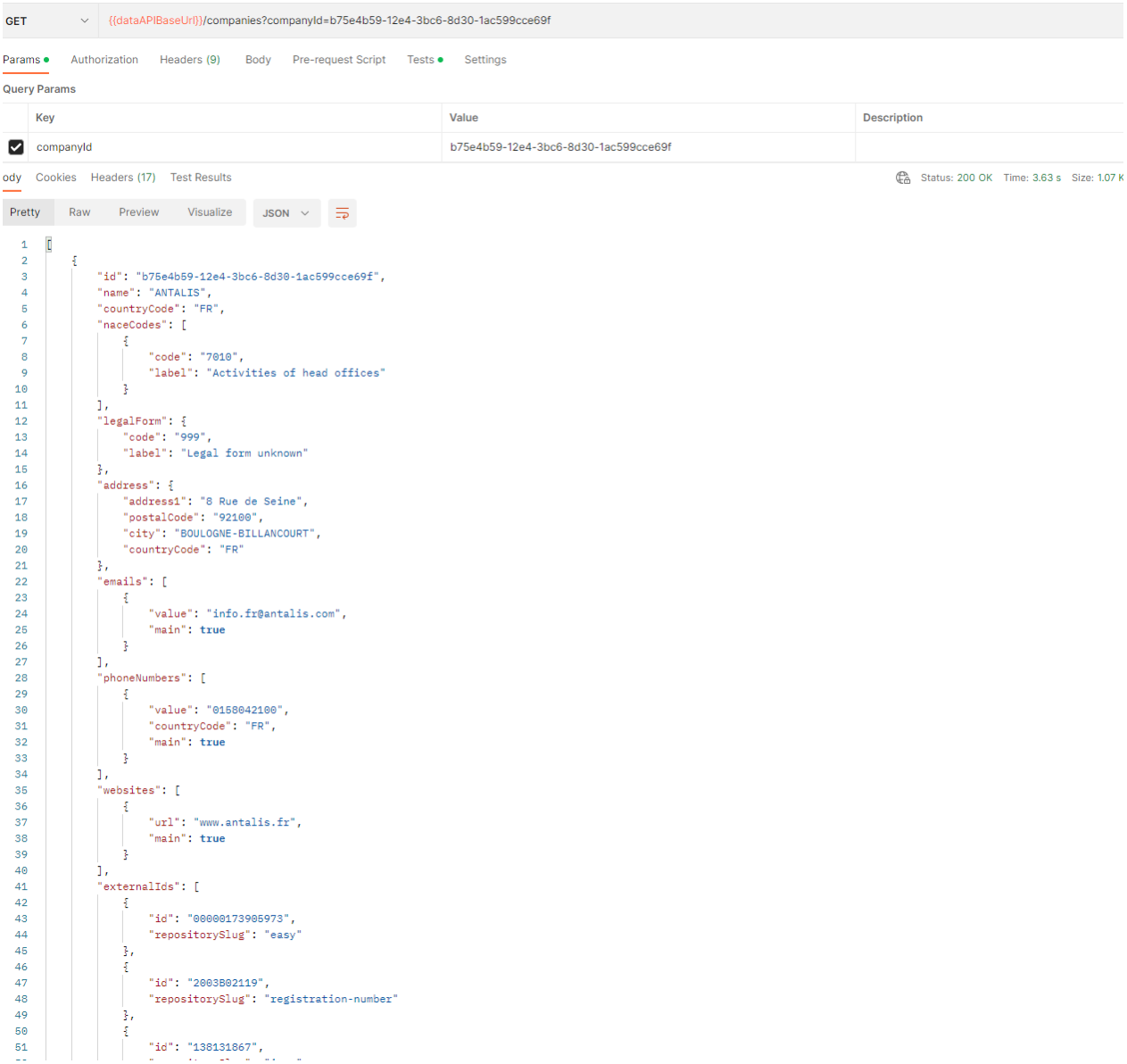


1.2.3 Search company by companyId

Allows to identify a company using the internal Coface id. This Id either is known without passing the precedent searches, either is returned when identifying the company as it could be seen in the previous methods.

If companyId parameter is provided, the other fields are ignored:

GET {{dataAPIBaseUrl}}/companies?companyId=b75e4b59-12e4-3bc6-8d30-1ac599cce69f



In all these cases, response should be identical. One company is uniquely identified.

Response contains information about the company requested: internal id inside ICON, name, countryCode, naceCodes, legalForms, address, emails, phoneNumbers, websites and externalIds.

Reaching this point, it means that customer either uniquely identified the company and can go to next step of checking the list of available products and order, or did not succeeded to find company.

1.3. Request a company identification.

In case you not managed to identify wanted company, you have the option to request to Coface a company identification and then to order needed product. You need to do it in 2 steps: first step is to request company identification and second step is to order a product for it. Please check below the 2 steps (sections 1.3.1 and 1.3.2)

1.3.1 Request the identification.

Endpoint to be used: ***POST*** *{{dataAPIBaseUrl}}/companies/identifications*

No parameters are required, but inside the request body, some information is required to be provided:

- at least one of the *externalID* needs to be provided OR
- for some countries where no legal identifiers are needed, it is enough to provide the *address.countryCode* in combination with: name, *address.postalCode* and *address.city*.

In case the provided information is not correct, response will be 400 Bad Request.

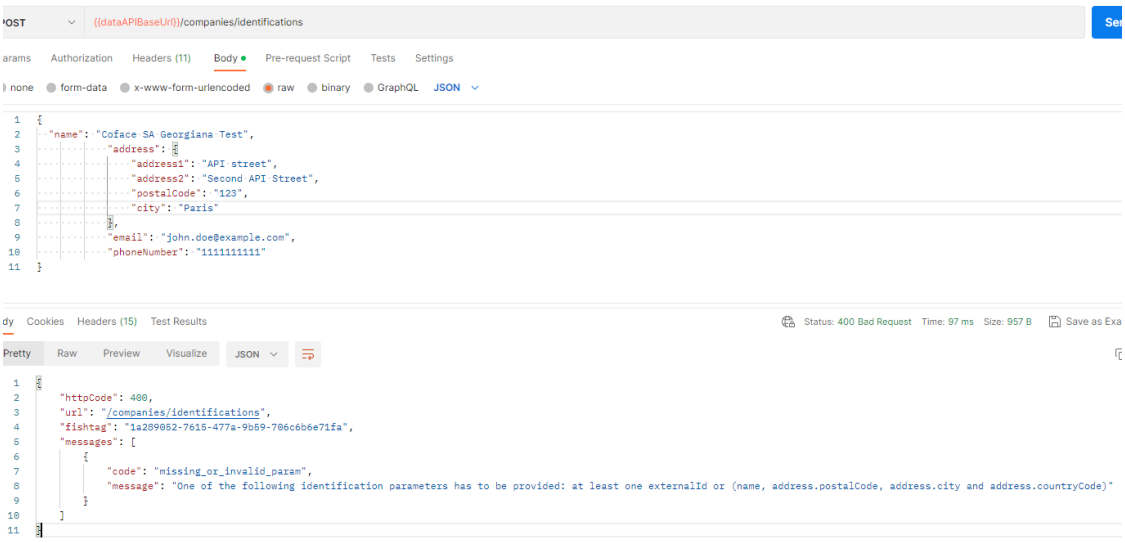
Remarks: Information about which countries and what legal identifiers need to be provided for each country, you can find via endpoint:

GET *{{dataAPIBaseUrl}}/companies/repositories?countryCode={XX}.*

For each identifier, there are 2 fields which show if is needed to be provided for a company identification or not and under which format:

"isMandatoryCompanyCreation": **true**,
"repositoryFormat": "VAT number or statistical number or registration number must be entered, there are no rules for r egistration number."

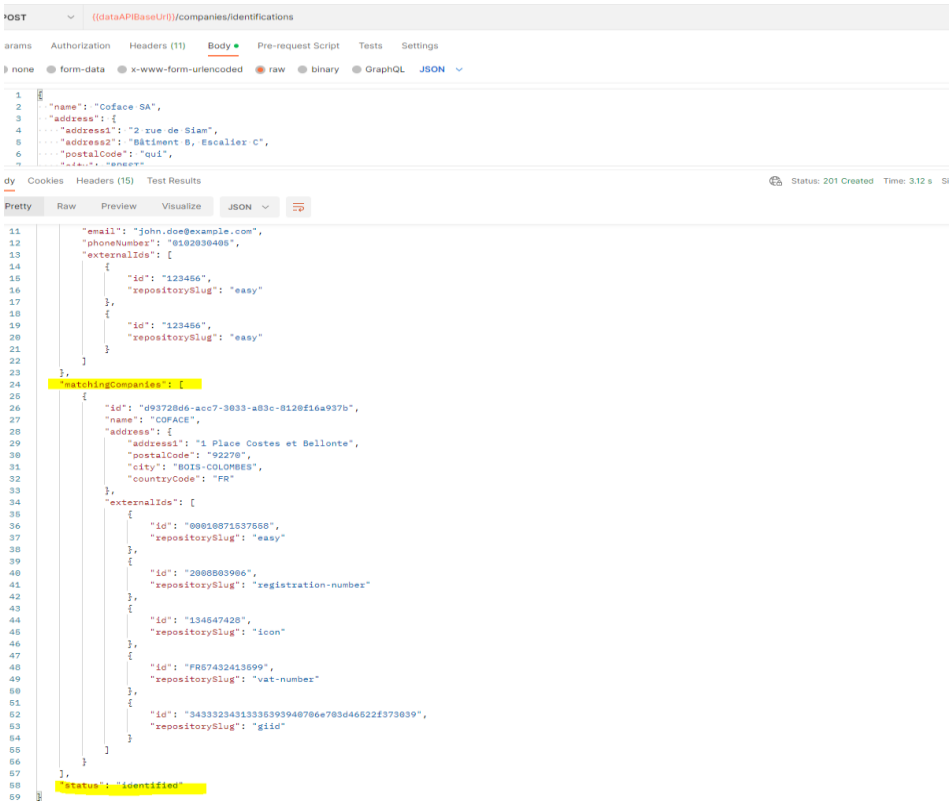
Error case: no address.countryCode



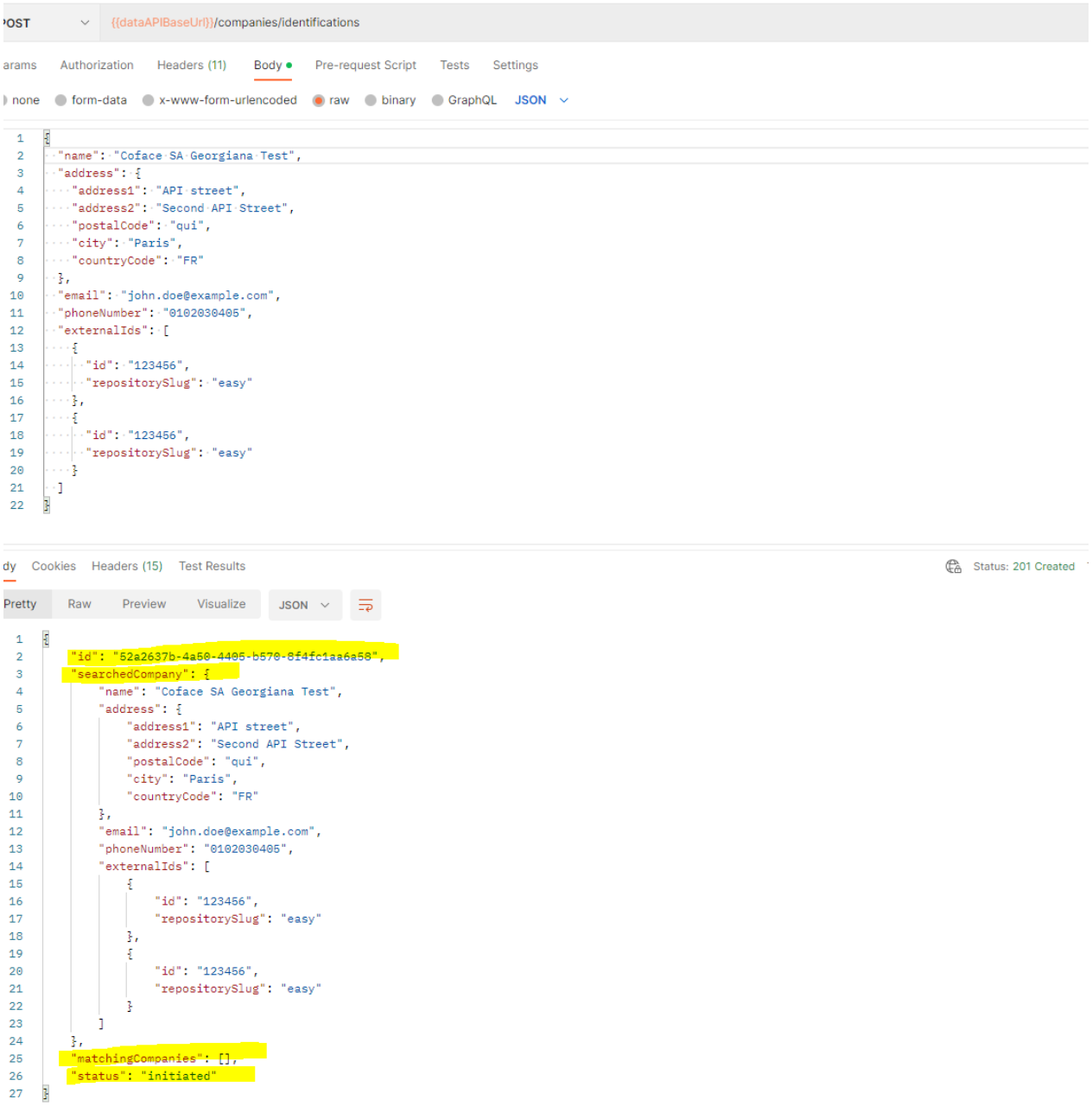
Success case- Request body:


```
{
  "name": "Coface SA",
  "address": {
    "address1": "2 rue de Siam",
    "address2": "Bâtiment B, Escalier C",
    "postalCode": "qui",
    "city": "BREST",
    "countryCode": "FR"
  },
  "email": "john.doe@example.com",
  "phoneNumber": "0102030405",
  "externalIds": [
    {
      "id": "123456",
      "repositorySlug": "easy"
    },
    {
      "id": "123456",
      "repositorySlug": "icon"
    }
  ]
}
```


If based on the information provided, ICON identifies the company, then the response will provide the info about searched Company in the field *matchingCompanies* together with status "*identified*" in the field "*status*":



If ICON cannot identify the company based on information provided, response will contain: *id* (which can be used for a future ordering), *searchedCompany* element containing company data provided in the request, *matchingCompanies* element which will be empty and *status* element with value" initiated".



1.3.2 Request the needed product.

The second step that needs to be done is to order the wanted product for this company, even if it is not yet identified.

See details under 1.4.2 Ordering a product for a company not identified.

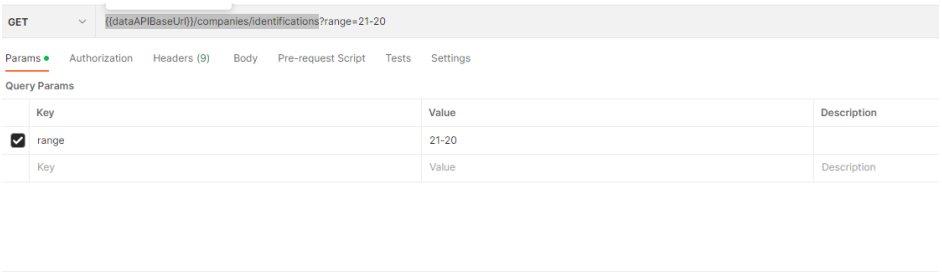
Remark!! Once you do also this step, the corresponding identification request is not appearing anymore as open, and it will not be returned anymore by any endpoint. The future status of the

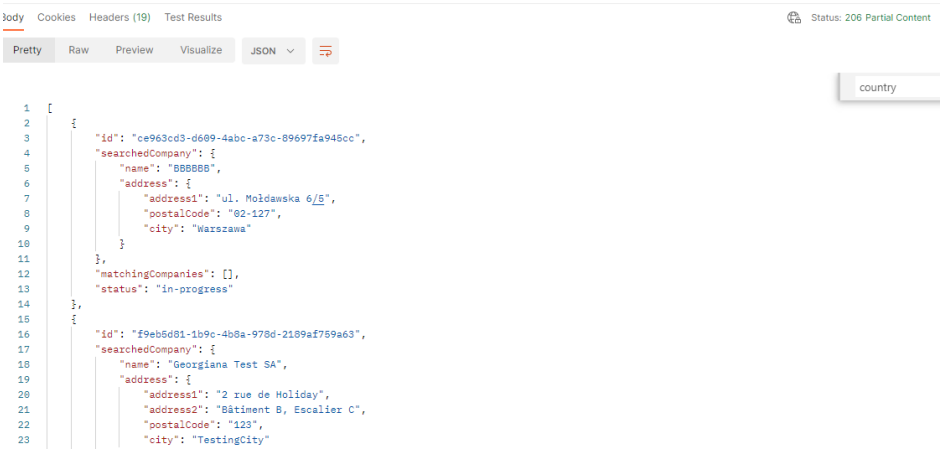
investigation is given actually by the status of the order placed on this step, ordering the wanted product. When order is finished, then either the company was identified, and product content can be downloaded, either the company could not be identified and product content it will be a so-called Negative report containing information about reasons of no identifications.

1.3.3 Request all identifications

All open requests to identify companies made by your profile are shown under endpoint:

GET {{dataAPIBaseUrl}}/companies/identifications





Please be aware that API is returning all **open** identifications, meaning all identifications which were not yet used (companyIdentificationId) to order a product with them.

Once you do also the second step, ordering a product via companyIdentificationId, the corresponding identification request is not appearing anymore as open and is not returned anymore by this endpoint.

1.4. Ordering a product

The structure of the request depends on the result of searching/identification of company.

1.4.1 Ordering a product for an identified company:

Remarks!!

1. **Slugs and product codes**

Attention: for DRAGON products (DRA, CO, AO) availability is not immediate. Therefore *processingTimeInstruction* option should have value **immediate_if_available_inquiry_otherwise** or **inquiry** with investigationSpeedLevel: normal.

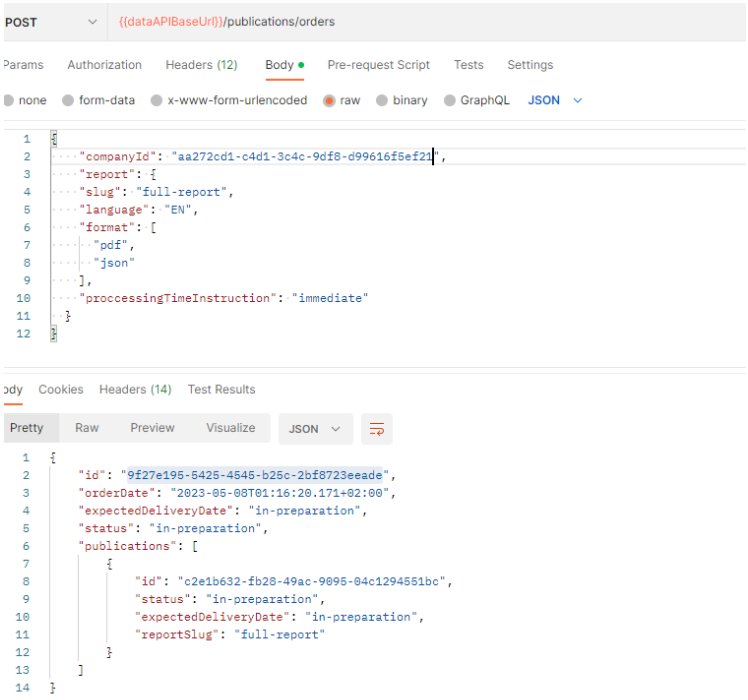
2. **Format of reports**

API allows to request report in different formats such as JSON, XML, PDF or HTML. For JSON and XML it will require additional mapping on customer side to integrate structured form of data to his own system needs.

3. **Ordering a product** can be made either using the internal *companyId* value returned by the API in one of the previous commands (the ones for identifying the company) OR using one of the ***externalIds*** (same, they are returned by the API when identifying the company)

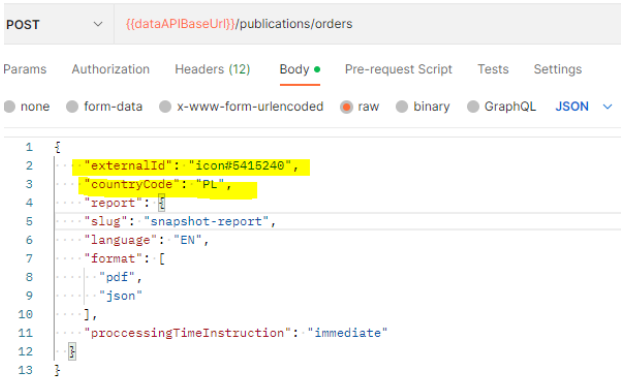
a. Ordering a Full Report via companyId:

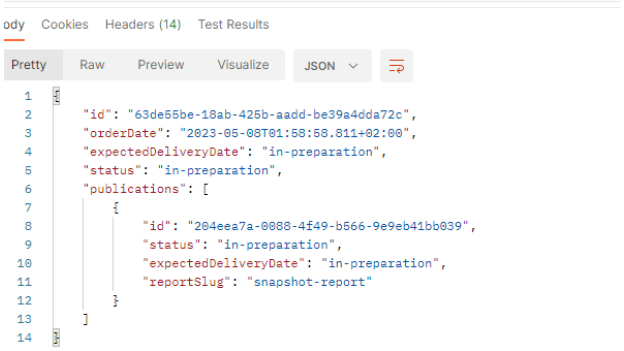
POST {{dataAPIBaseUrl}}/publications/orders



b. Ordering a Snapshot report (light report) via externalId:

POST {{dataAPIBaseUrl}}/publications/orders





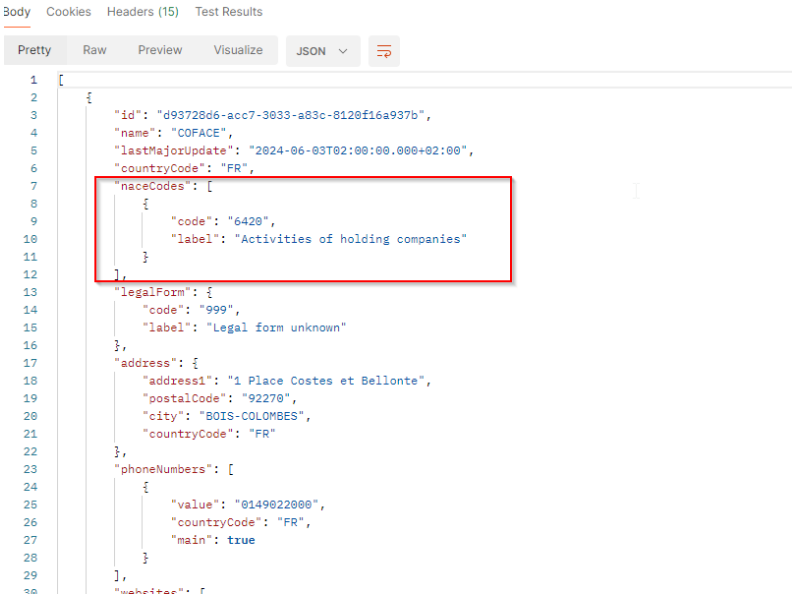
c. Ordering a Sector Risk Assessment

When ordering a SRA product (Sector Risk Assessment) you need to consider the following:

- If in the moment of placing the order, we do have already in ICON database information about the sector of the company, then you can place the order without mentioning the sector/nace code. If this is the case it can be checked if naceCode information is present when identifying the company.
- If in the moment of placing the order, we do not have such information, then:
  - either you can provide in the body of the request the corresponding sector,
  - either a business report needs to be ordered so that we can have in our database the corresponding sector
  - otherwise, the order will be rejected, as we would need to protect against a massive number of orders which would need to be worked manually.

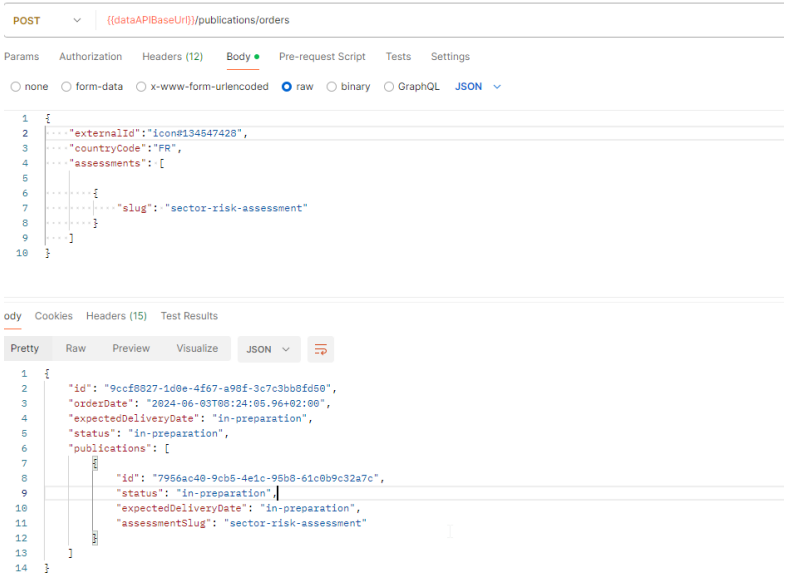
Checking if company has already Nace Code in ICON database:

GET {{dataAPIBaseUrl}}/companies?countryCode=FR&externalId=icon%23134547428

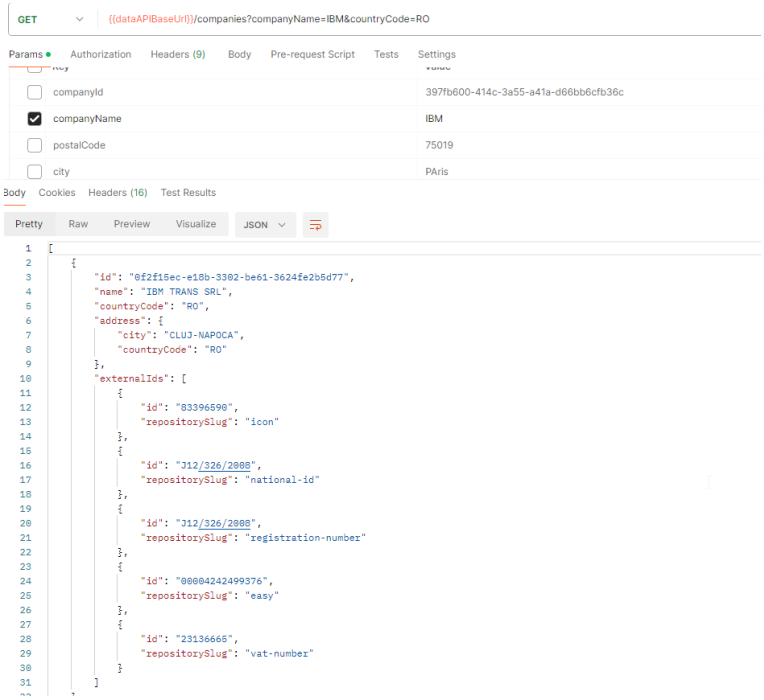


For this company, **ordering a SRA is possible without providing the NaceCode inside the body of the request**:

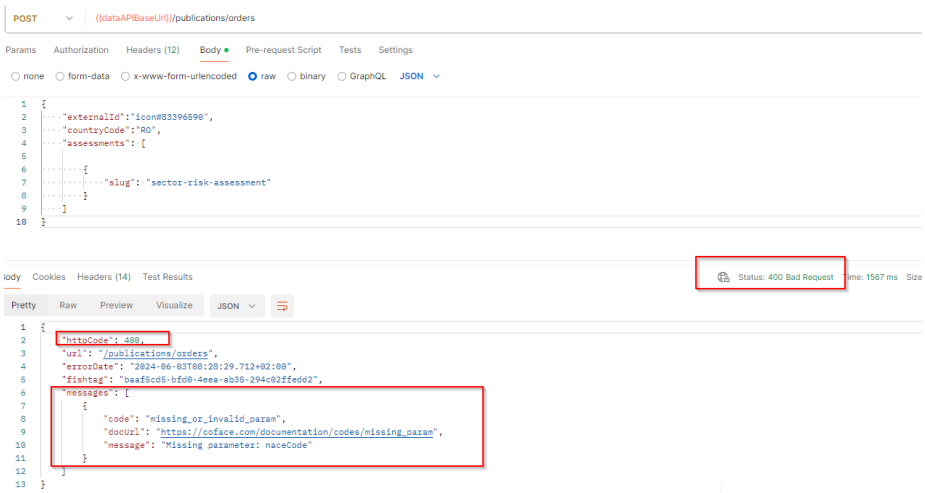
POST {{dataAPIBaseUrl}}/publications/orders



Sample of company which does not have Nace code in our ICON database in the moment of ordering:

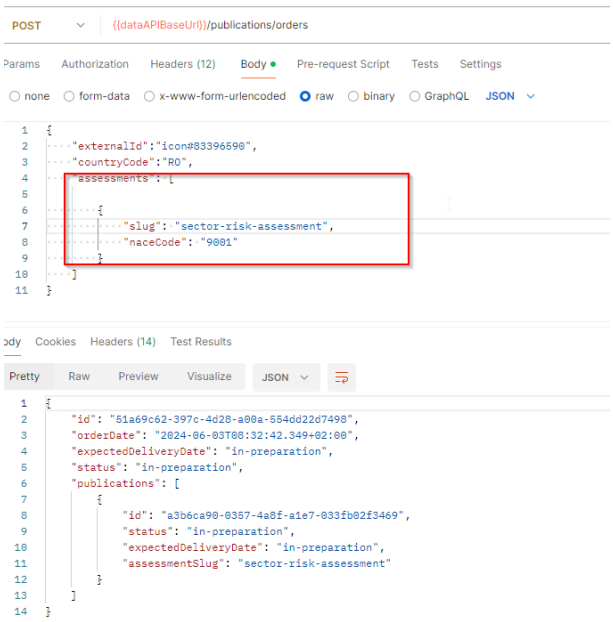


If we would want to order a SRA for this company, without providing the sector, order will be rejected:



Now there are two options:

- Either we order a SRA product mentioning the sector of the company in the body of the request:



- either a Full Report is ordered, so that we will obtain the naceCode and save it in our database. If after ordering a Full Report, naceCode is known and received, it will be saved into our database, and then a SRA can be ordered.
- But it could be also cases, that even after ordering a Full Report, naceCode remains unknown. Then for sure the order is still rejected. We could not provide a Sector assessment if we do not have the Sector.

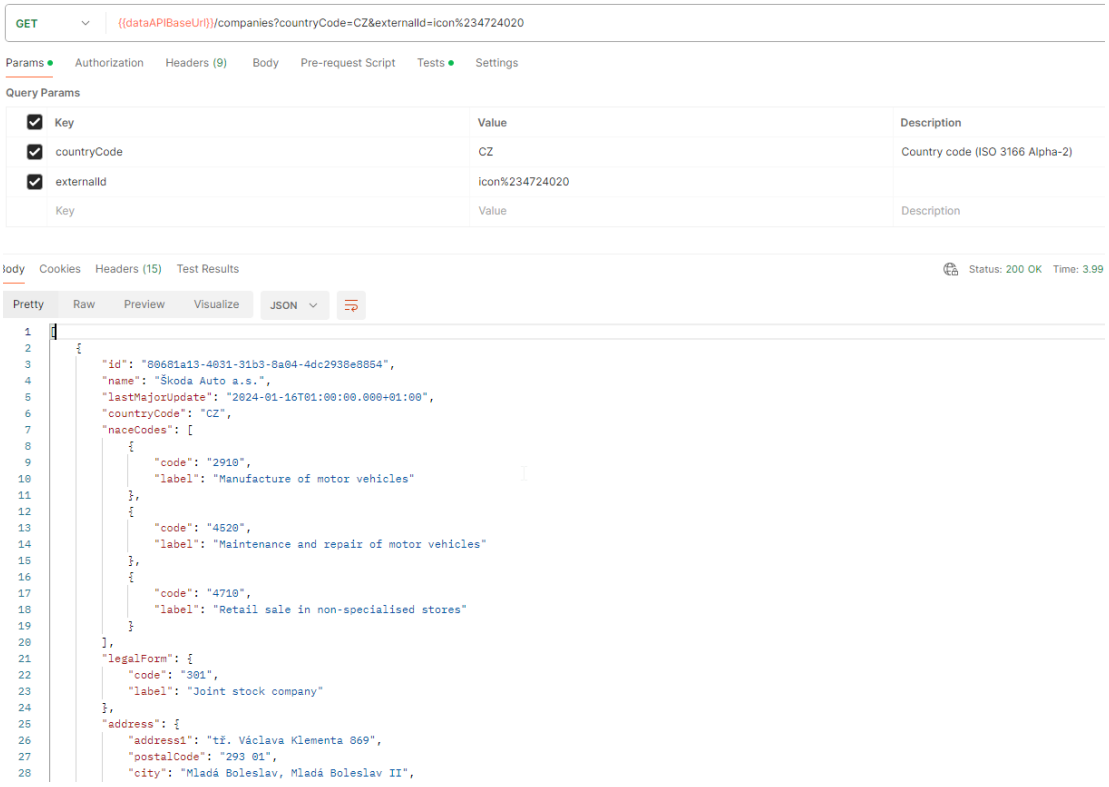
d. Ordering Financial products

When ordering financial products please consider that initial steps for searching and identifying the company should be performed also here. Please check section 1.2 Search a company (identification of a company in the database) for more details.

If you identified your company you can order via one of the possibilities mentioned : via companyId, or via one of the externalIds, iconNo or VATNo, etc.

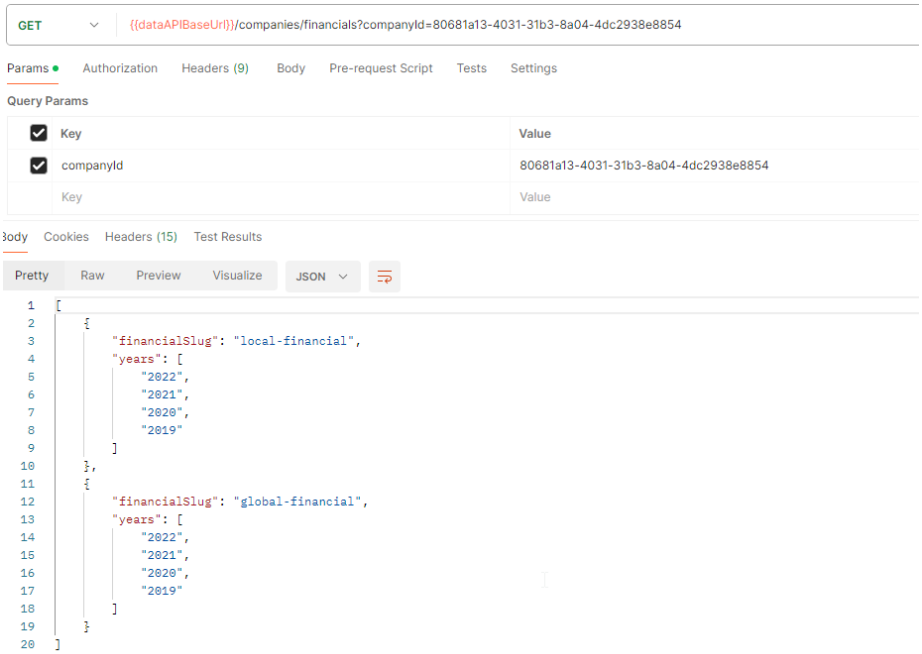
We will treat here the case that we identified the company, and we have the iconNo:

GET {{dataAPIBaseUrl}}/companies?countryCode=CZ&externalId=icon%234724020



The most important step is to check if ICON do have the financial information available for this company and which are the years. This can be done via products endpoint. Here we will use the companyId from above response:

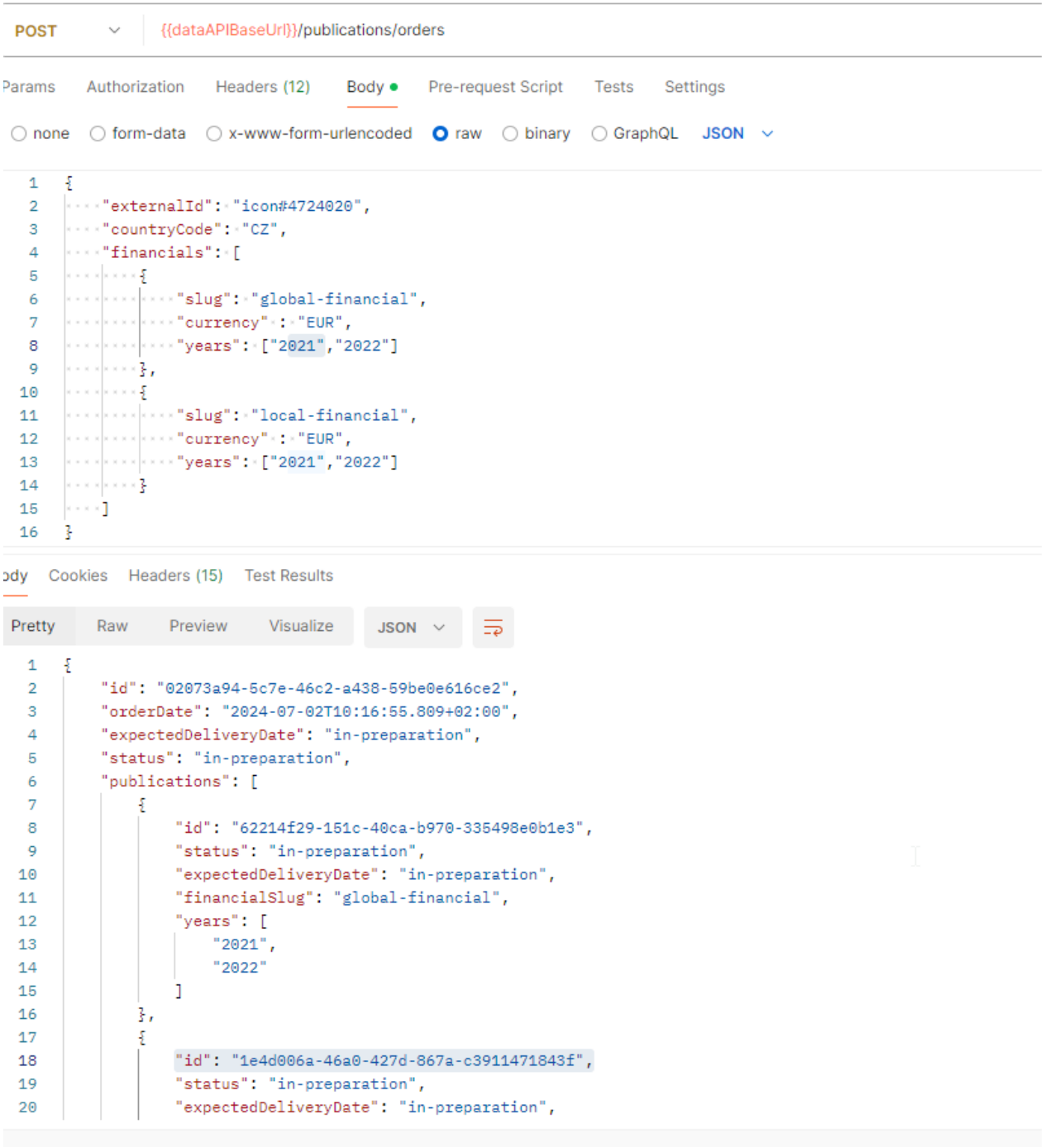
GET {{dataAPIBaseUrl}}/companies/financials?companyId=80681a13-4031-31b3-8a04-4dc2938e8854



Response will contain the years for which the financial information is available.

Now, we will order via one of the externalId, and we will choose iconNo:

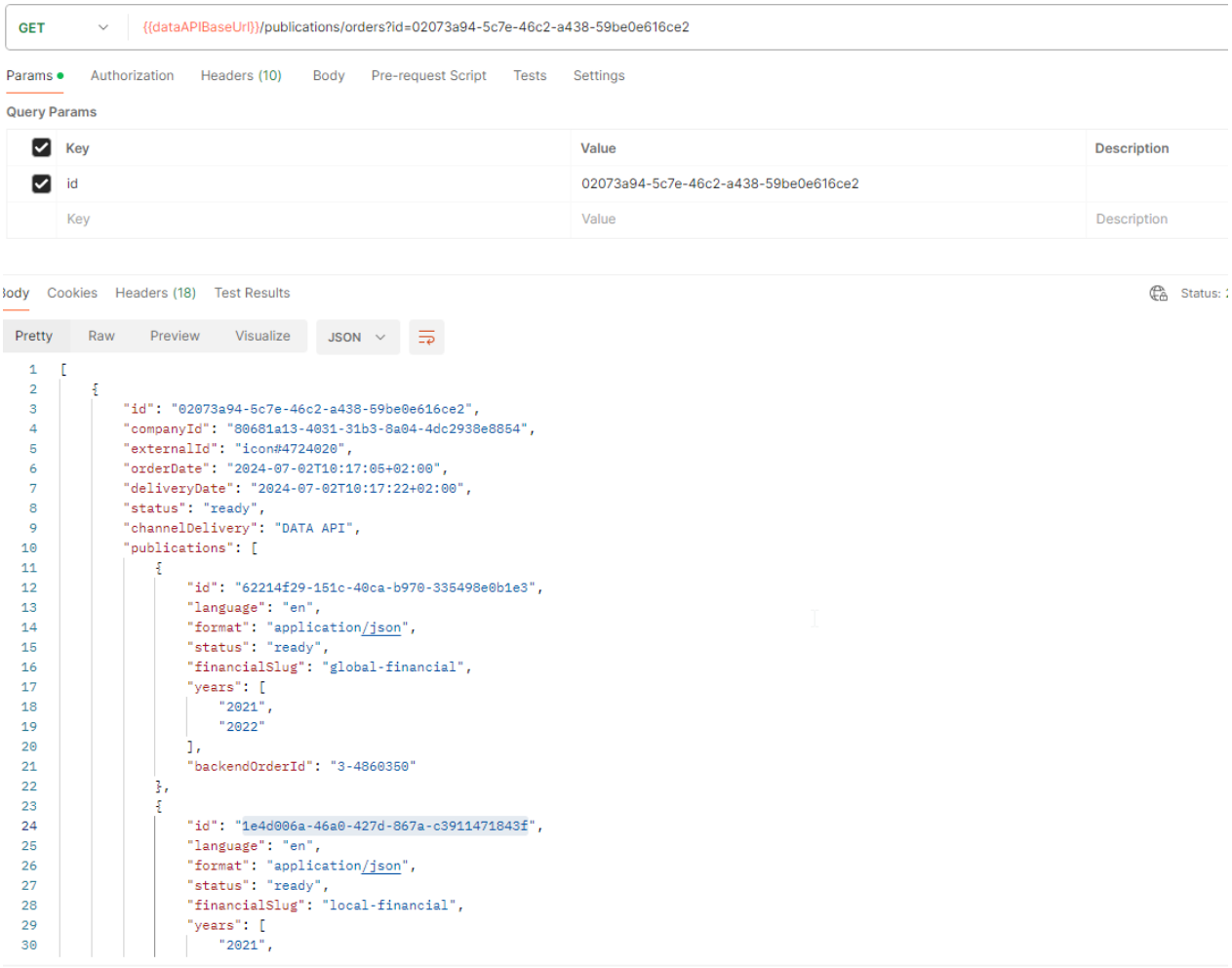
POST {{dataAPIBaseUrl}}/publications/orders



Request body should contain the "*financials*" section where the "*slug*", "*currency*" and the "*years*" values must be specified as in the above sample.

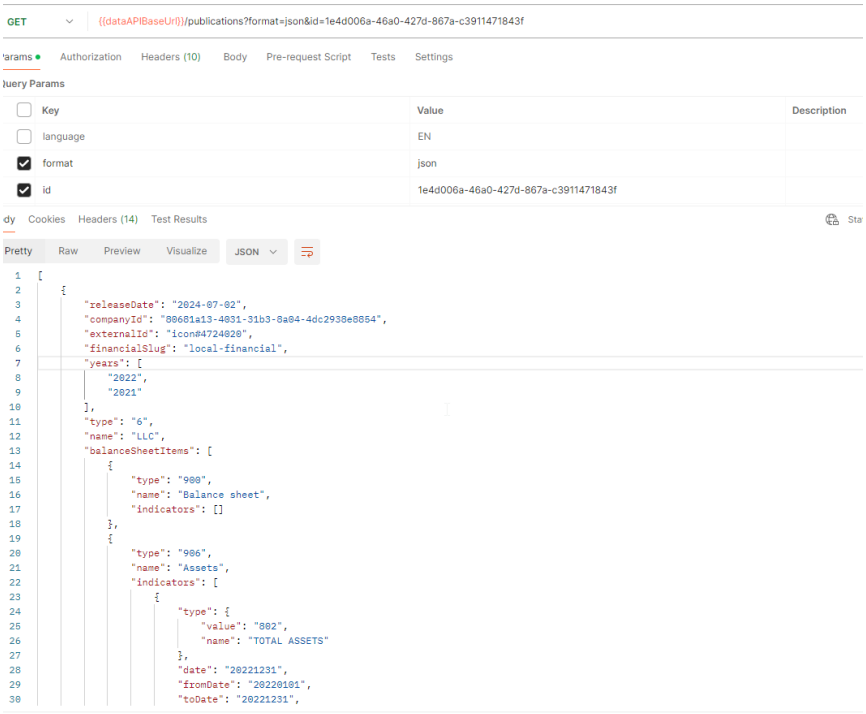
Response is containing the global id and ids of the orders placed, as for all the other products.

Now with the global id, can be checked the status of the order:

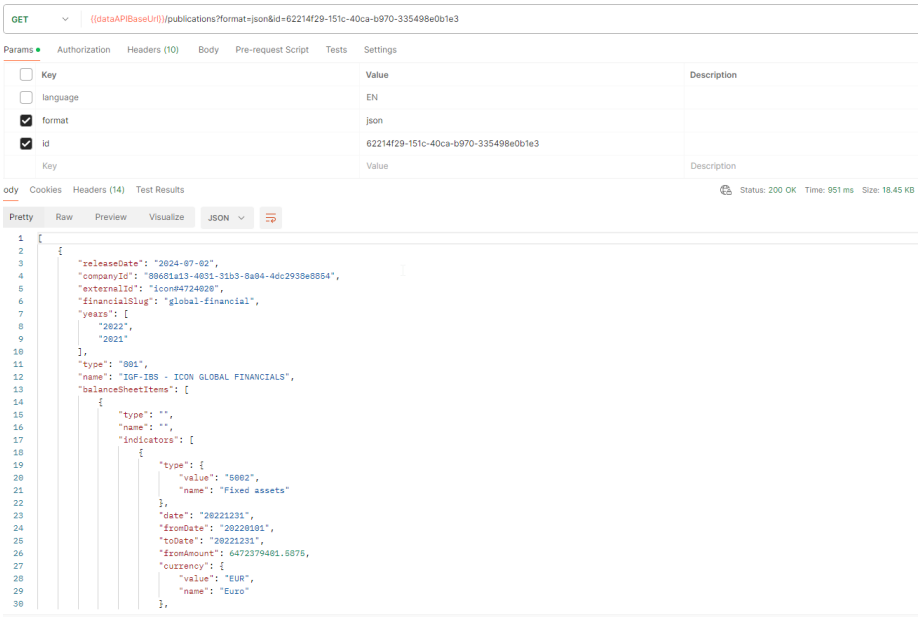


And with the publication ids, the content of each financial product can be downloaded.

Local financials:



Global financials:



Format of financial is always JSON. It is not mandatory to order both financial products, only one can be ordered. In POST request then, user will put only the wanted slug.

1.4.2 Ordering a product for a company not identified.

Prerequisite: you need to request an Identification of the company for which you need to order (See section **1.3.1 Request the identification)** so that you have the companyIdentificationId returned by ICON.

Endpoint to be used:

POST *{{dataAPIBaseUrl}}/publications/orders* or

POST *{{dataAPIBaseUrl}}/monitorings/orders*

Request body:

```
{
   "companyIdentificationId": "52a2637b-4a50-4405-b570-8f4fc1aa6a58",
   "report": {
   "slug": "full-report",
   "language": "EN",
   "format": [
     "pdf",
     "json"
   ],
   "proccessingTimeInstruction": "immediate"
  }
}
```

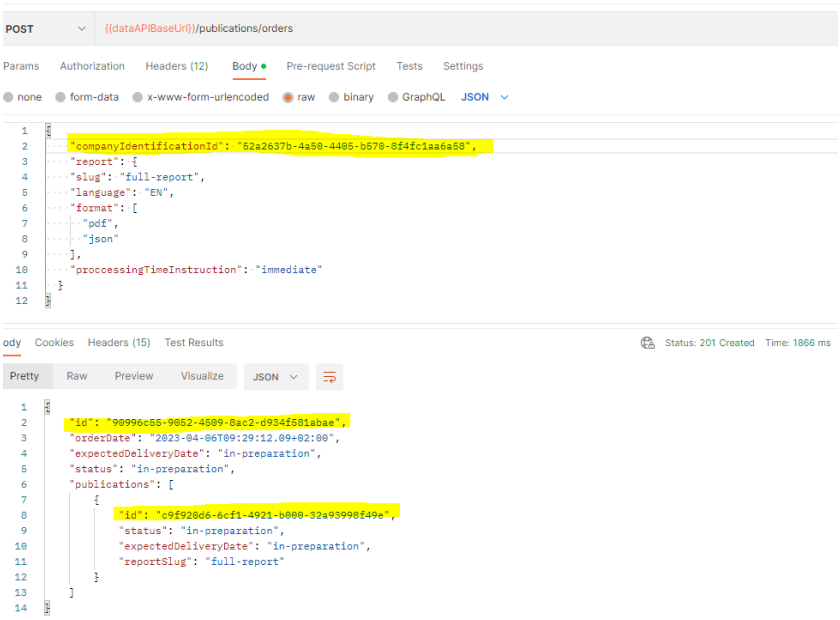
No parameters

Body of the request should contain mandatory information: *companyIdentificationId*, product section, which can be "*report*" or "*assessments*" or "*financials*"

More details and explanation for fields that need to be present for each of the section, can be found behind each element from swagger, by clicking on Model and after that, on arrow down for each drop down that is available.

Swagger can be found under AWS Dev portal: https://devportal.coface.com/apis/qves74kqv0/v1

Response is the same as when ordering for an identified company. From here, all the other steps can be performed: get status of the order and, when ready, get the content of the order.



1.4.3 Ordering a product using Callback mechanism.

Callback mechanism is provided by Coface API in case the order was not finished immediately. Therefore, we offer the customer the possibility to be announced when the order is completed.

Ordering a product using the callback mechanism has exactly the same steps as described under section [1.4.1 Ordering a product for an identified company]

The only difference is the fact that in the body of the request one more field is added, ***callbackUrl.*** In this field, an URL address is given as input by the customer. "***callbackUrl"*** field should be placed outside of any of the sections "report", "assessments" or "financials".

More details and explanation for the field can be found inside swagger, by clicking on Model and after that, on arrow down for each drop down that is available.

Swagger can be found under AWS Dev portal: https://devportal.coface.com/apis/qves74kqv0/v1

How it works: We have a "callback" job which is running on each minute, and for all orders which are finished by that moment, and they were marked as orders with callback(based on the value from http request), we sent one ping to the corresponding URL. We do not provide anything to customer server via that call, no info about the order or about the product. It is just a ping to URL provided, and we will retry several times (16 times) if the URL is not available. We are not having any authentication method when calling the URL.

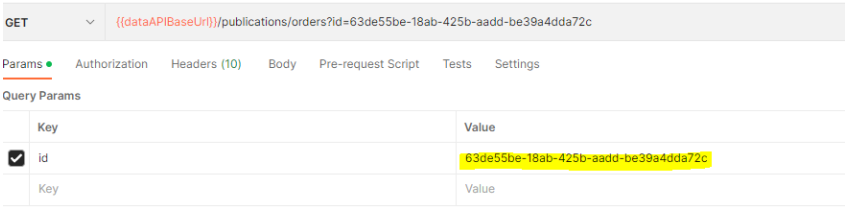
When customer receives a ping on the corresponding URL, it can access the corresponding Endpoint of the API to download the content of the product.

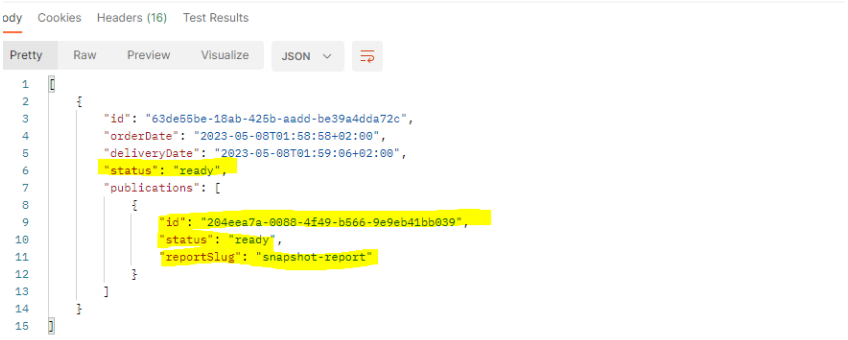
1.5. Checking order status

After using the POST method, customer gets as response a JSON structure containing:

- one id which allows checking the status of the order (initial status of request is “in-preparation”)
- “publications” section containing all products ordered in same order. All of them have this initial status “in-preparation.”

In order to check the status of the whole order, first id value must be used for endpoint *GET /publications/orders?{id}*





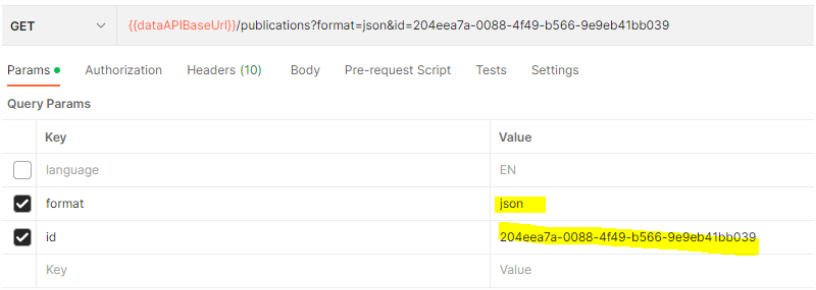
Status of the order will be ready if all publications of the order are ready. If one of the publication is not ready, the status of global order will be still “in-preparation”.

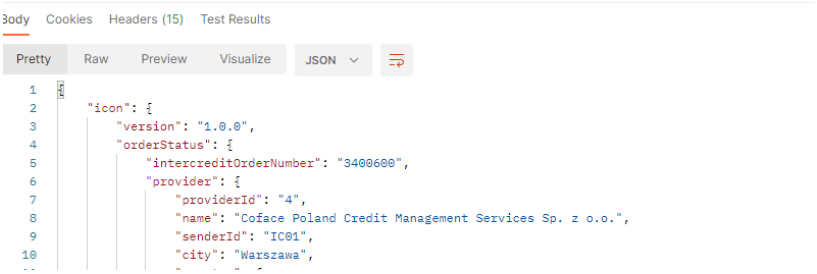
Attention! Only when status of the order / publication will be “ready” you will be able to retrieve the content of the corresponding product ordered (publication)

1.6. Retrieving the data

If order is finished, customer can call method *GET /publications?format={format}&{id}*

Format and id parameters are mandatory. Format should be the one used when POST call was made and id should be the id returned in “publications” section returned by API on POST method.





1.7 Request Monitoring

Customer can request monitoring, by **ordering one of the combined products**: *Full Report with Monitoring, Snapshot Report with Monitoring, DRA with Monitoring, @ Credit Opinion with Monitoring, Advanced Opinion with monitoring, Country Risk Assessment with monitoring, and Sector Risk Assessment with monitoring.*

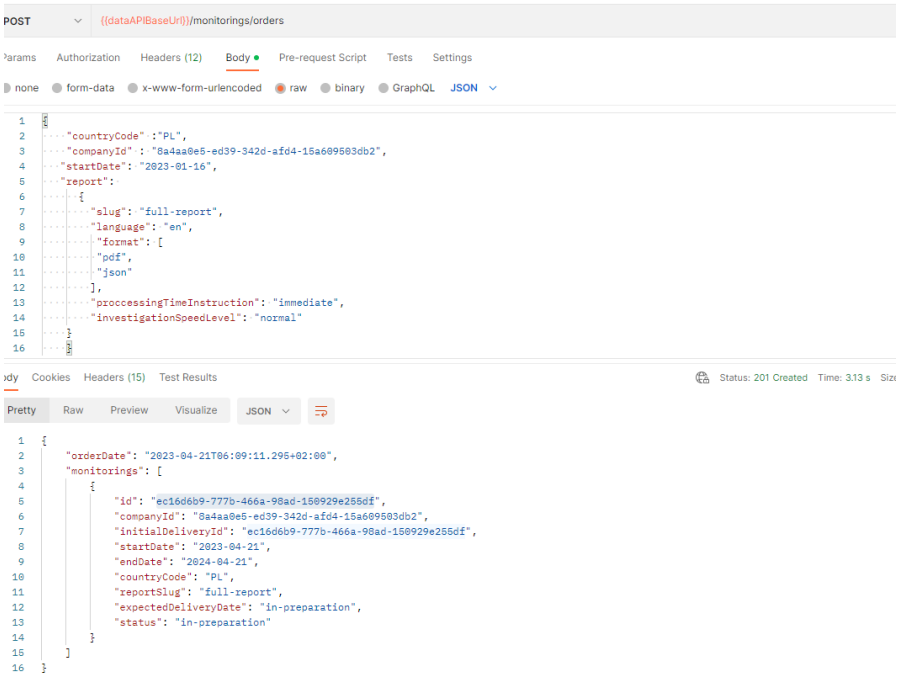
Only identified companies can be monitored.

By default Monitoring should last 12 months.

In order to order a combined product, endpoint ***POST*** *{{dataAPIBaseUrl}}/monitorings/orders*

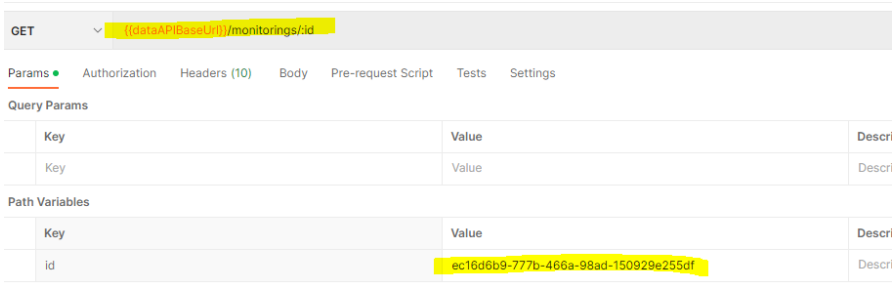
needs to be used, using the slug/product id returned by API when company was identified.

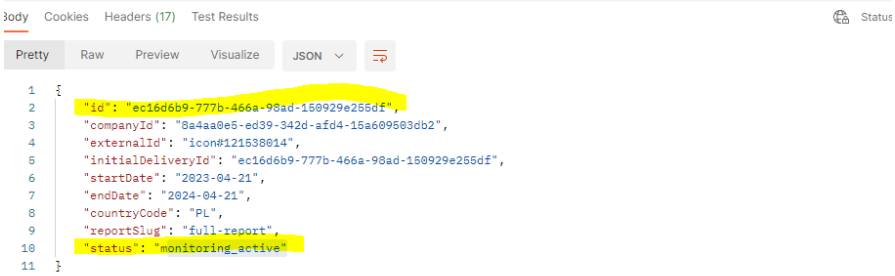
So, if I identified a company and I have the companyID, I am placing the monitoring order:



In the response I am getting some information like: Id of the monitoring order, start and end date of the monitoring, status (which is “in-preparation”) , expectedDeliveryDate, companyId, country Code and product for which monitoring was ordered.

Now, I can check the status of this order using endpoint ***GET*** *{{dataAPIBaseUrl}}/monitorings/:id*





If the request is successful (status is “monitoring_active”) then the company starts to be monitored.

1.8 Retrieving content of the report / product ordered via monitoring order

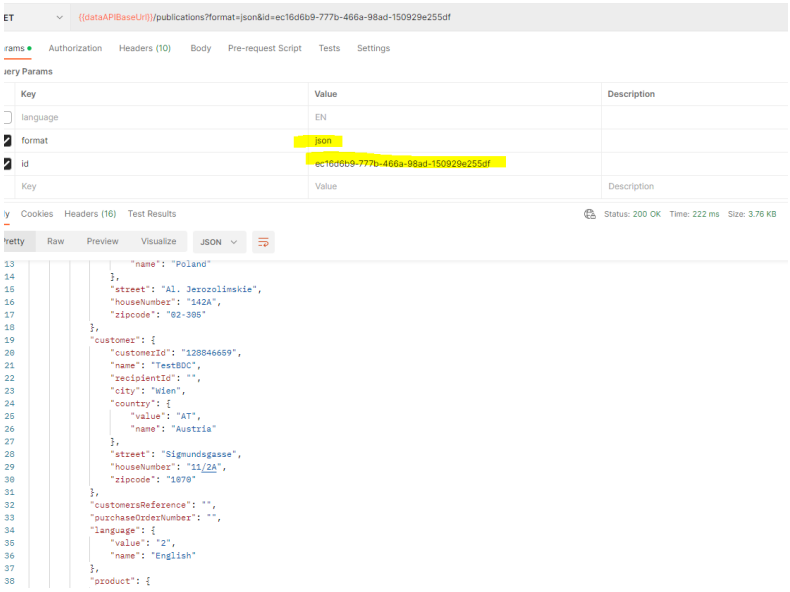
Because a combined product was ordered, we can also retrieve the content of the product ordered.

Using the id returned by the POST monitoring order (see 1.7), we are getting the content of the product using endpoint ***GET*** *{{dataAPIBaseUrl}}/publications?format={}&id={orderId}*

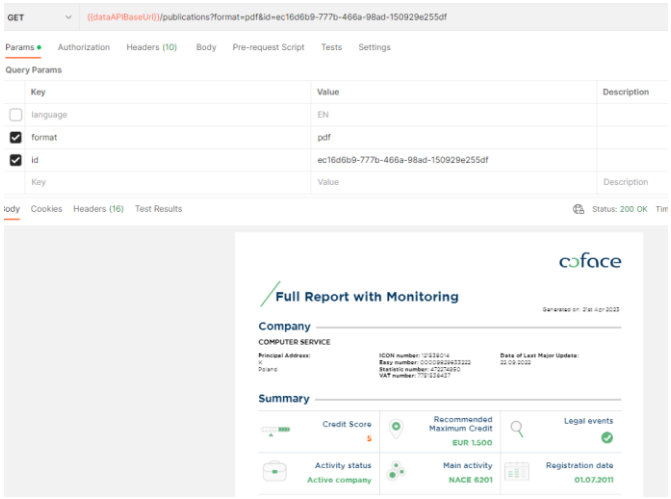
Where orderId is the id returned the API when posting the monitoring order

Format parameter is mandatory. As we ordered in 2 formats, we can retrieve both of them (reminder: only for CCE companies)

JSON format:



Pdf format:



1.9 Retrieving monitoring notifications.

As a customer which ordered monitoring (you have companies monitored) you will need to fetch the changes which are happening on these monitored companies.
There are 2 ways of doing this: using Callback mechanism or pulling regularly the notifications. The only difference between the 2 ways is that, if for the first way you will perform actions to get the notifications only after ping is sent to provided URL, for the second method, you are doing it without any trigger when you want, depending on your need. You can do it regularly daily, weekly, etc.

First part of Callback mechanism is done when ordering: inside POST body request you need to provide an URL where we will ping when a notification is created.

Please check here, how to configure a notification callback. (TBD)

Remarks:

- If you have this mechanism configured, then in the moment a notification is created on our side, for this order, the provided URL will receive a ping from Coface side.
- Ping call does not push any information (content or meta-data about the order) and does not have any authentication mechanism.
- Notifications are created and pushed once per day

If you used callback mechanism when ordering, then you will do the below steps triggered by the ping received in the provided URL.

If you did not used callback, then is up to you to implement a scheduler to do actions described below, depending on your need.

Steps to be done for getting the notifications (independent if you used callback mechanism or not):

Step 1: check via notification Endpoint which notification was received. Call can be made using several parameters in order to have a more feasible response. It can be done using data filter (to obtain all notifications per day or per specific period) OR via following filters:
product/order/company/customerReference/channelDelivery

Requests:

Getting all notifications daily:

GET *{{dataAPIBaseUrl}} /notifications?deliveryDateFrom=2024-07-29&deliveryDateTo=2024-07-29*

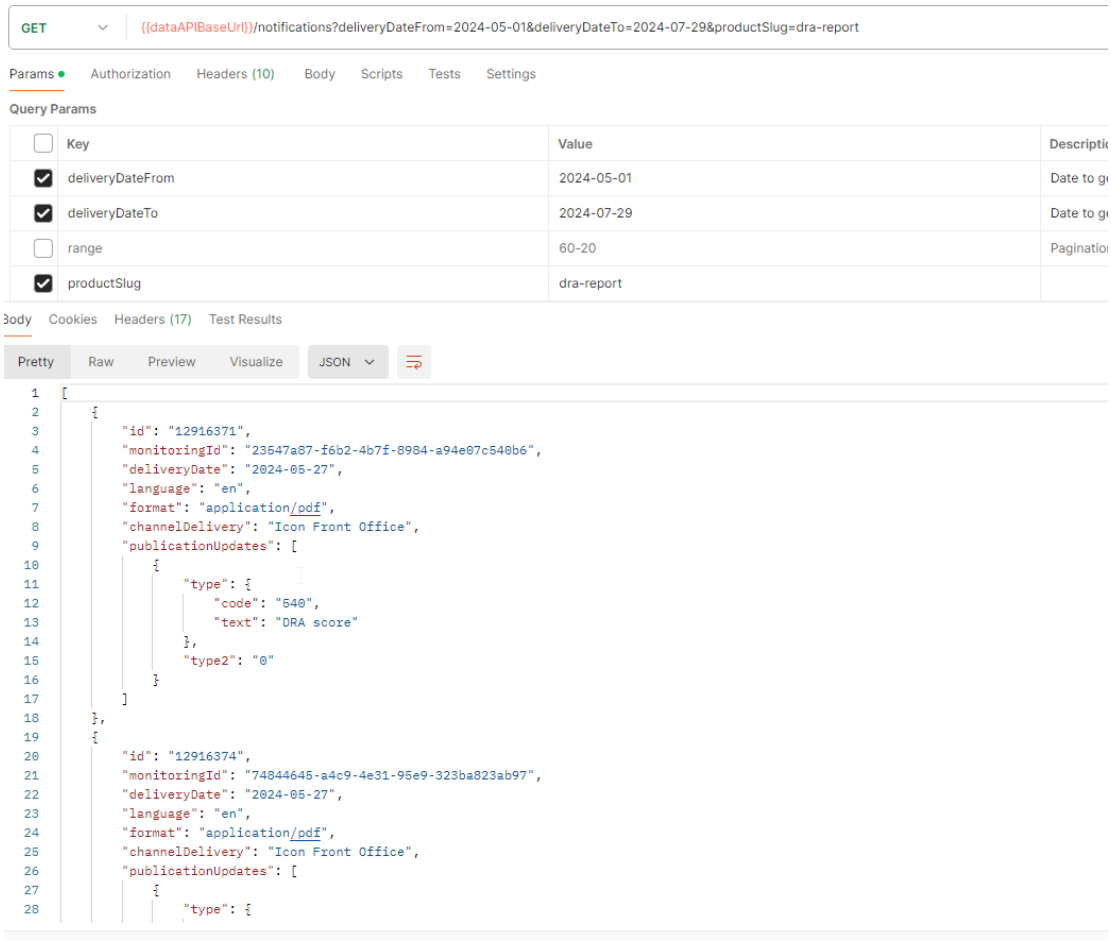
Getting all notifications for a specific product, any company, since the beginning of the monitoring:

GET *{{dataAPIBaseUrl}}/notifications?productSlug=dra-report*

Getting all notifications from a product in a specific period of time:

GET *{{dataAPIBaseUrl}}/notifications?deliveryDateFrom=2024-05-01&deliveryDateTo=2024-07-29&productSlug=dra-report*

Response:



Contains an array of all notifications requested. Number and type of notifications depend on filter used in request. It can have all notifications for one product but for all companies, for all products all companies, or other combinations of filters.

Each notification will have: id of the notification, id of the monitoring order, deliveryDate, format, language, channelDelivery, callbackUrl and publicationUpdates object.

The content of publicationUpdates object, depends on which product the notification belongs to. We can have the whole new content of the product or just some highlights of what have been changed.

Step2: For each notification received, take the id and the format of the notification, and get the content of the notification via corresponding endpoint.

Request:

GET *{{dataAPIBaseUrl}}/notifications/:notificationId/content?format=json*

Response:

{

"monitTitle": "Company monitoring details",

"monitReleaseDate": "",

"monitDetailsProduct": "DRA Monitoring",

"monitDetailOrderNo": "17291221",

"monitCpDetailsName": "Company CORPORATION",

"monitCpDetailsAddress": "100 PARK AVE, 07932-1049 FLORHAM PARK, US",

"monitCpDetailsContact": "",

"monitCpDetailsICONNo": "123456789",

"monitCpDetailsVATNo": "",

"monitCpDetailsRegistrationNo": "000489930",

"monitCpDetailsLegalStatus": "",

"monitCpDetailsStatisticNo": "",

"monitDRAValue": "8015",

"monitDRAChangesOldValue": "8013",

"monitDRAProbabilityOfDefaultNewVal": "New POD 0.4%",

"monitDRAProbabilityOfDefaultOldVal": "Previous POD 1.3%",

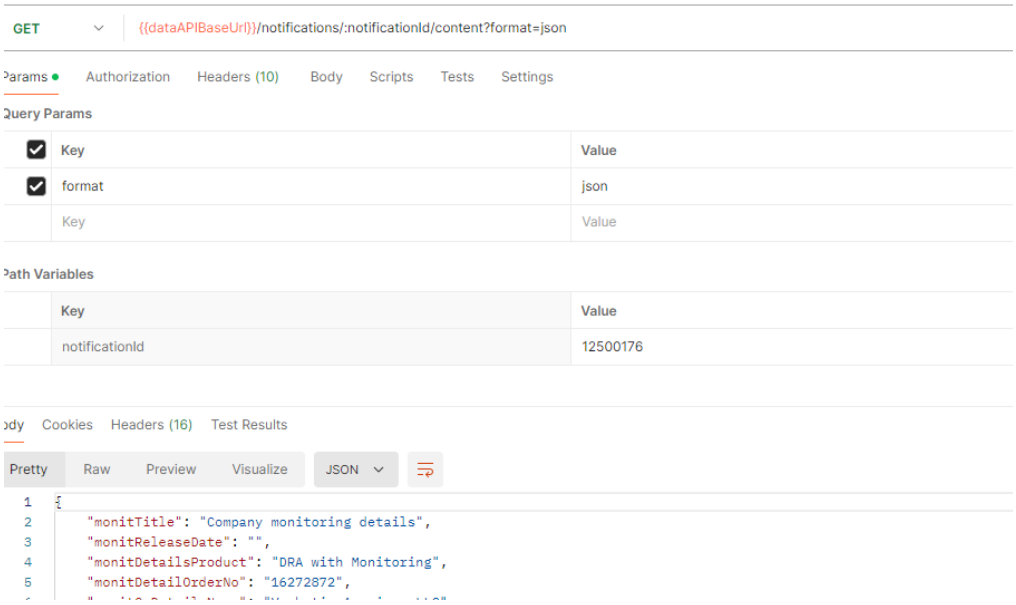
"monitProductCurrency": "",

"isUrbaOrder": **false**,

"isApionDragonProduct": **false**,

"internationalCompanyName": ""

}



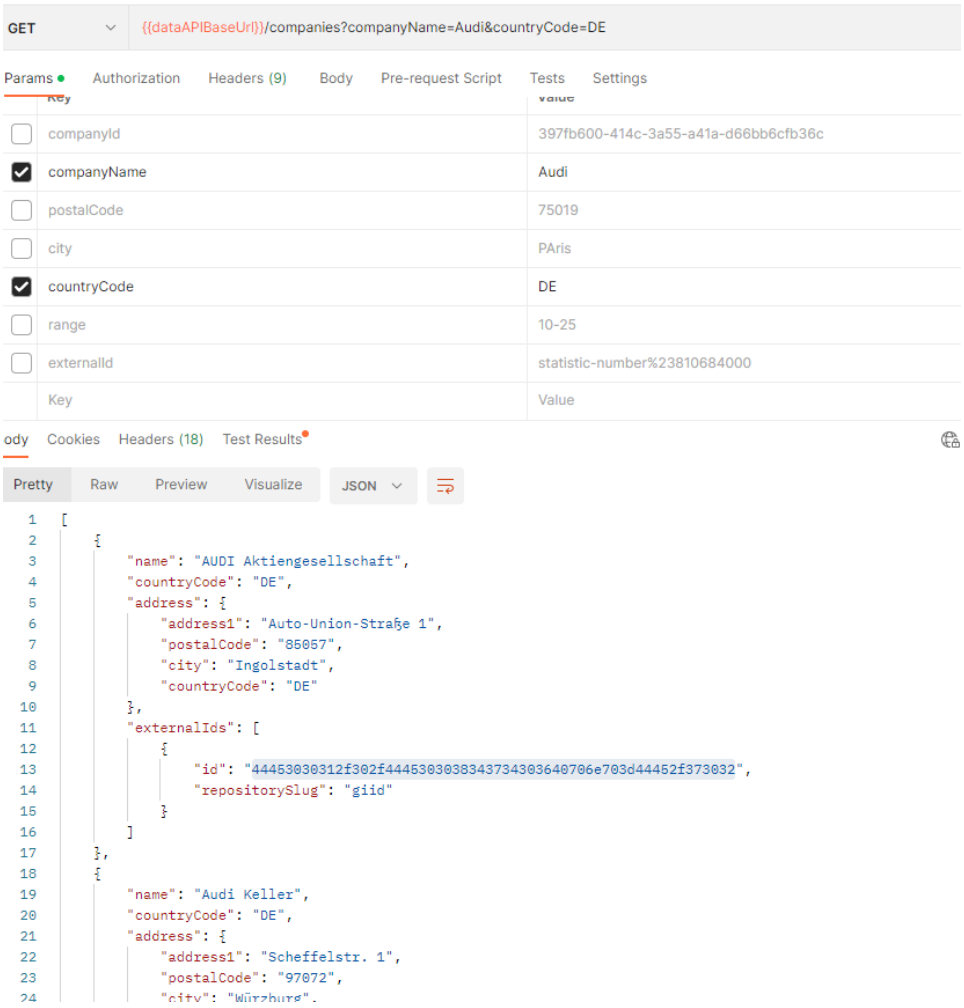
2. Use cases

2.1 Ordering multiple products for same company

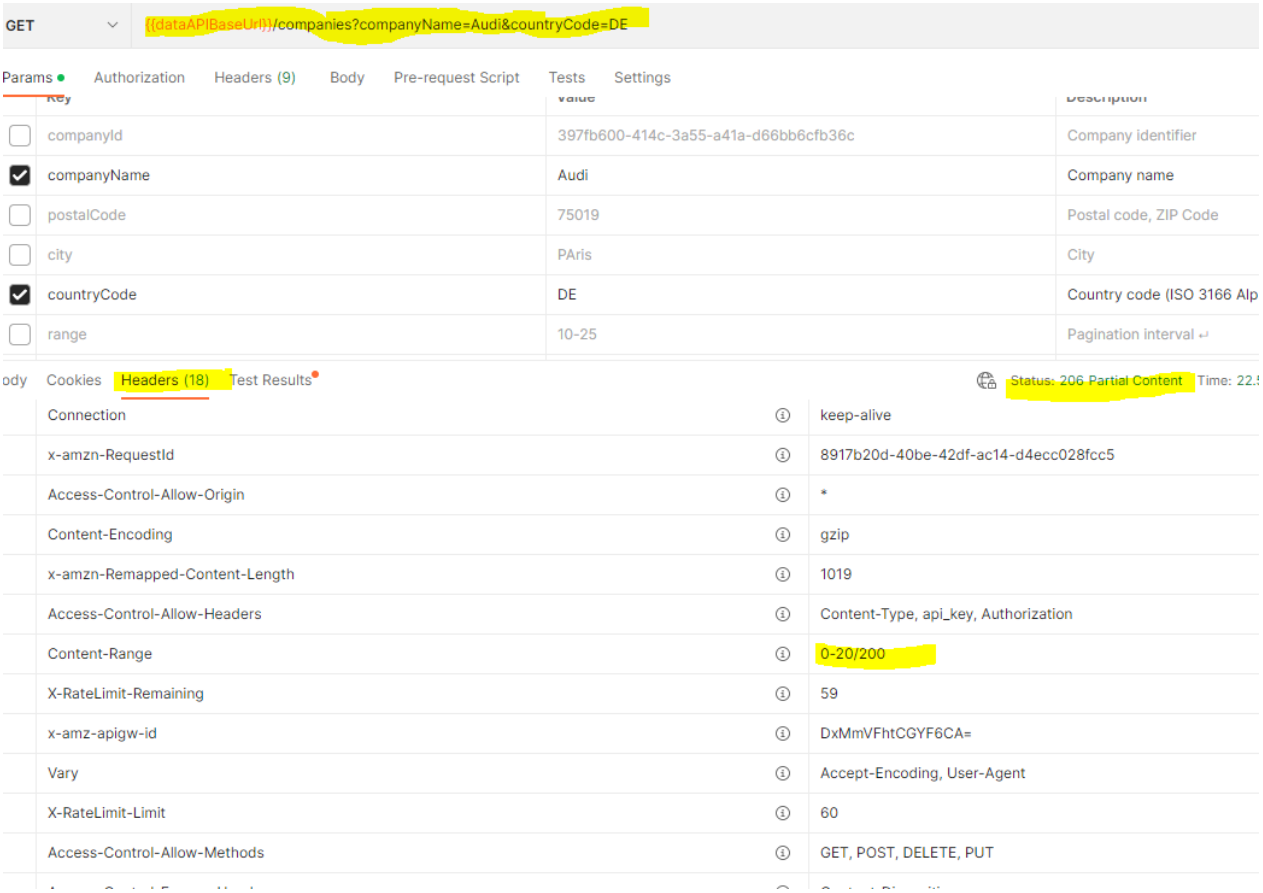
2.1.1 Ordering 2 assessments in one call

You want to order some products for Audi Company in DE

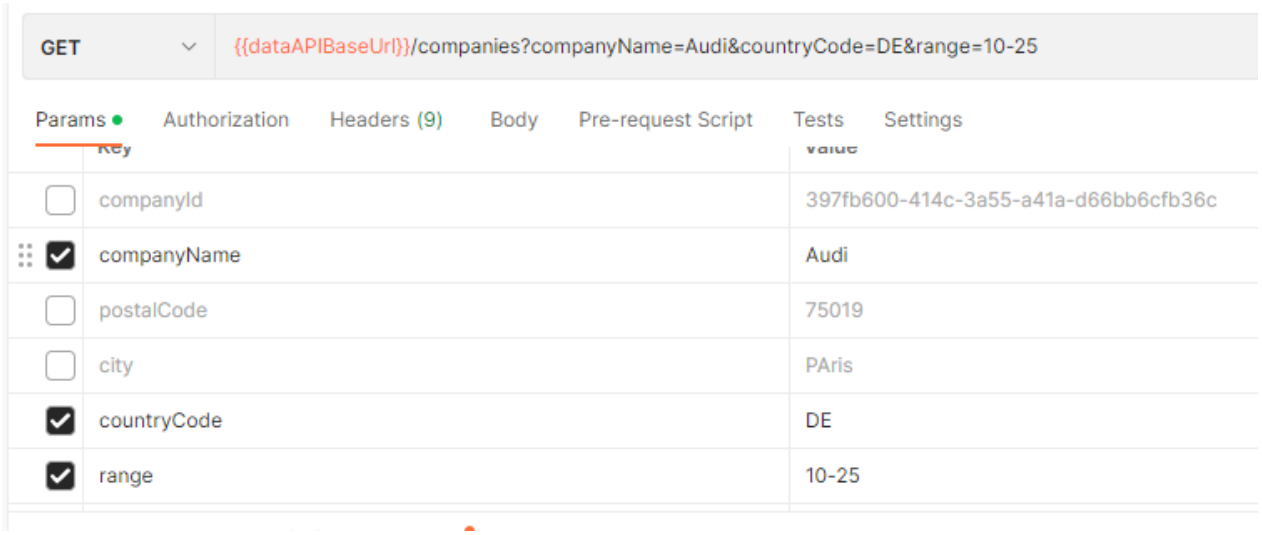
First step is to search, using company name and country code, as this is what you know currently:



Call is sent to one of our providers (the one to which your customer profile is routed to). Provider is returning all results that it has with this name and country code. In this case, API returned 200 results.
Per page we have 20 results, and pagination is in place (you can see the 206 partial content, and headers values highlighted below):

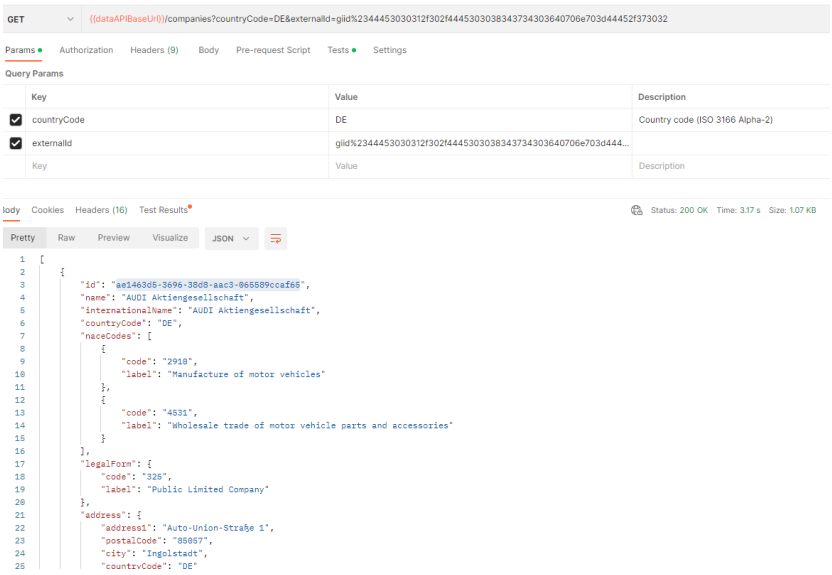


You have 200 results. In order to check others which are not in first page, you need to use the range parameter. We are showing max 20 results per page.



Now, out of 200 companies, you need to identify one of them in order to order some products.

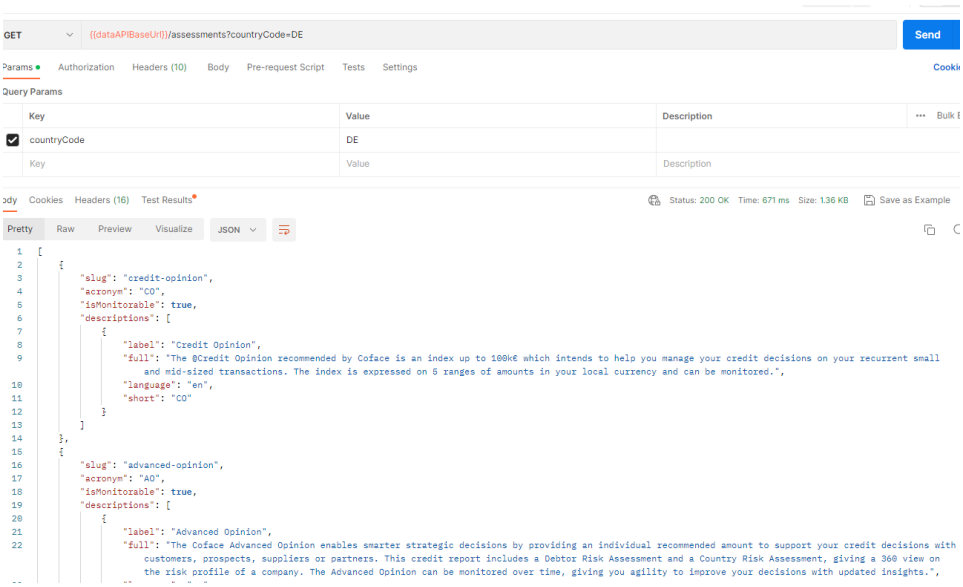
There are 2 ways: either you choose one, and take the giid value and use it as externalId parameter and make the order with it, OR, you make one more call and uniquely identify the company and then you can order using another identifier replied by API (vat number, companyId, icon no)



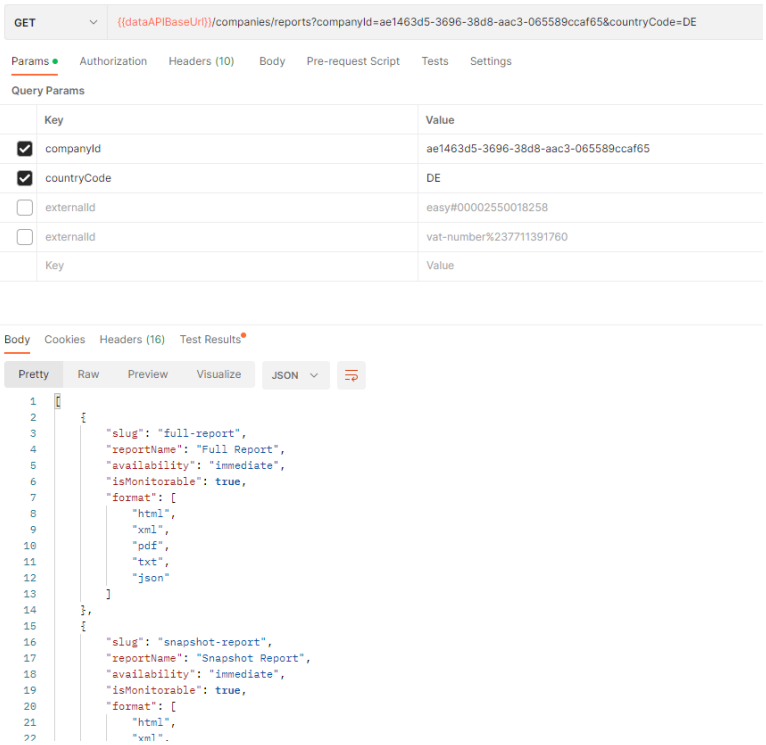
Now, company is saved into ICON, and has more identifiers associated: easy, icon nr, companyId, etc.

In this moment, you need to check and see what the available products are for this company, together with their availability (immediate or on-order):

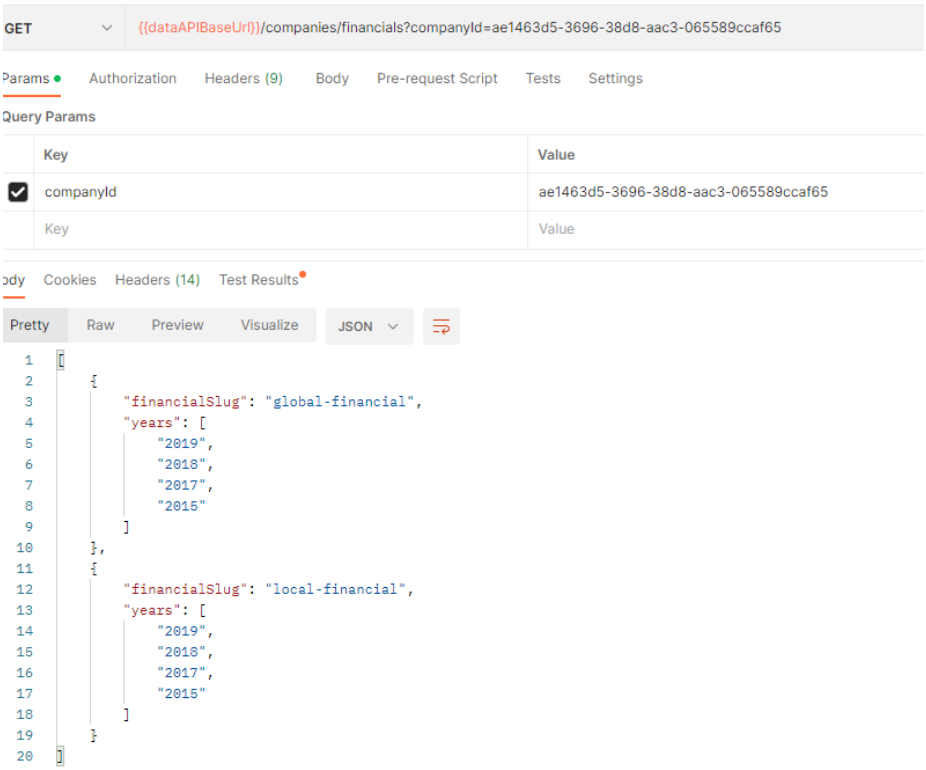
Check assessments:



Check reports:



Check financials:



Now is time to order:

Ordering 2 assessments: Country Risk Assessment and Sector Risk Assessment in same call:

POST {{dataAPIBaseUrl}}/publications/orders

Params Authorization Headers (12) Body Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary GraphQL JSON

```
{
    "companyId": "{{companyId}}",
    "assessments": [
        {
            "slug": "country-risk-assessment"
        },
        {
            "slug": "sector-risk-assessment"
        }
    ]
}
```

Body Cookies Headers (14) Test Results

Pretty Raw Preview Visualize JSON

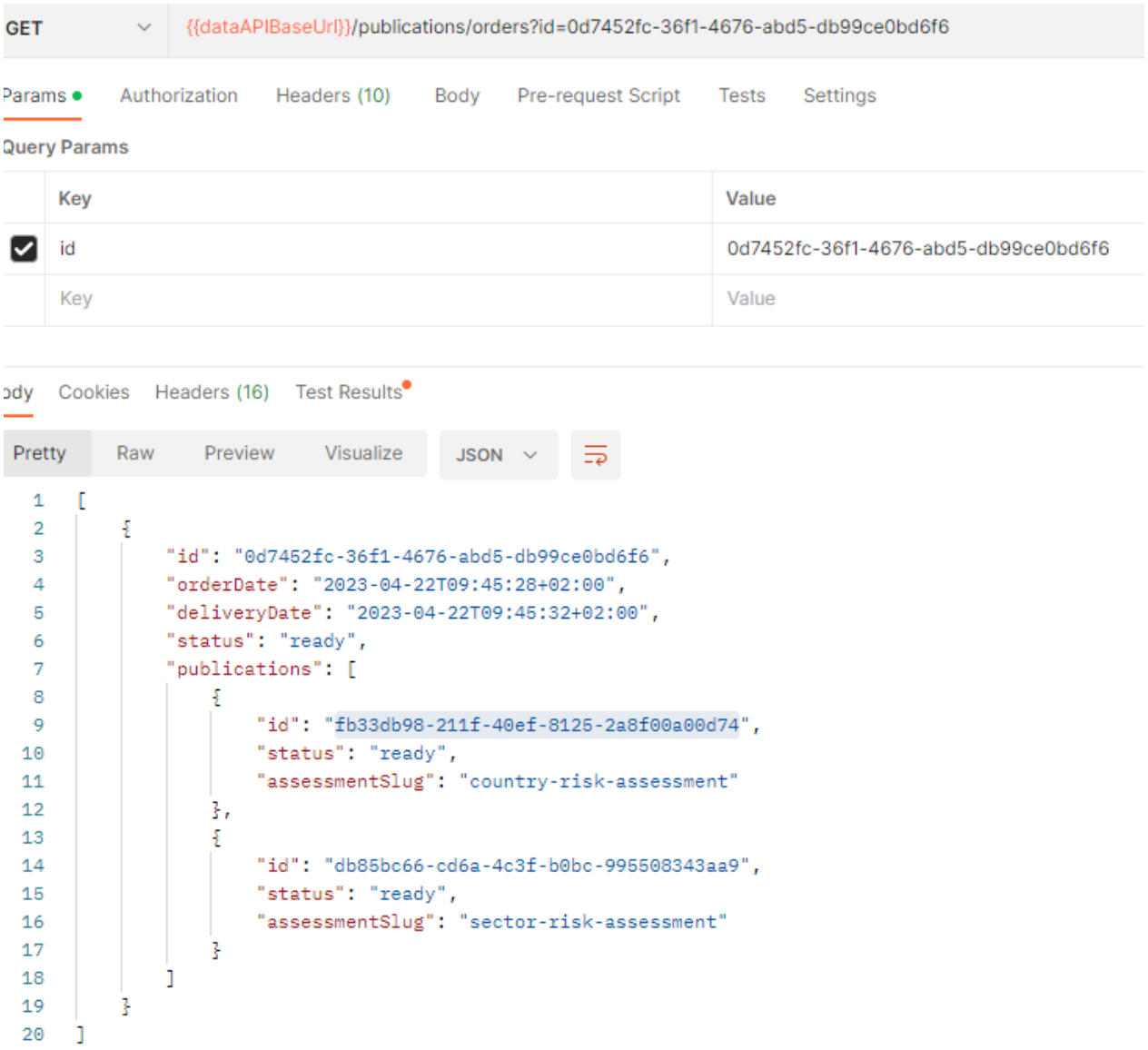
```
{
    "id": "0d7452fc-36f1-4676-abd5-db99ce0bd6f6",
    "orderDate": "2023-04-22T09:45:28.751+02:00",
    "expectedDeliveryDate": "in-preparation",
    "status": "in-preparation",
    "publications": [
        {
            "id": "fb33db98-211f-40ef-8125-2a8f00a00d74",
            "status": "in-preparation",
            "expectedDeliveryDate": "in-preparation",
            "assessmentSlug": "country-risk-assessment"
        },
        {
            "id": "db85bc66-cd6a-4c3f-b0bc-995508343aa9",
            "status": "in-preparation",
            "expectedDeliveryDate": "in-preparation",
            "assessmentSlug": "sector-risk-assessment"
        }
    ]
}
```

As we ordered 2 products in one order, in the response, we will get one global id of the order, first id from json and 2 ids corresponding to each product ordered.

Response of the POST call, will return status “in-preparation” if request is correct on first validations. But even if API returned Ids and “in-preparation” status, it might be that due to some technical or business reasons the order to not be created at all in the backoffice or to be cancelled.

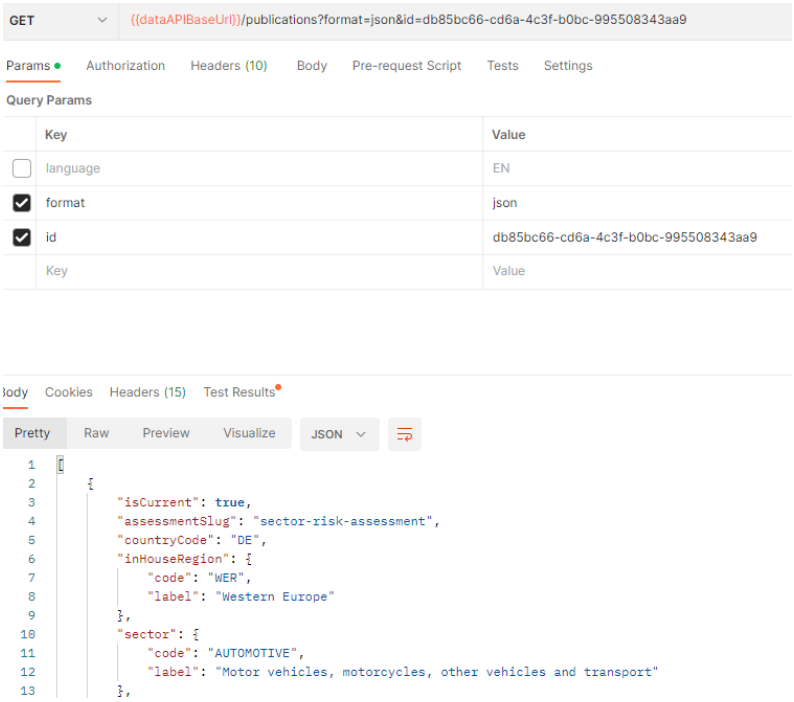
If , for example we order a product with instructions :"immediate" but the product was not available immediate, the call of POST order returns ID and status "in-preparation" in first phase, but when trying to call the status of the order, it can be cancelled, or answer can be an empty array, as order was not at all created in backend.

This is why we have the status call as mandatory step after ordering::

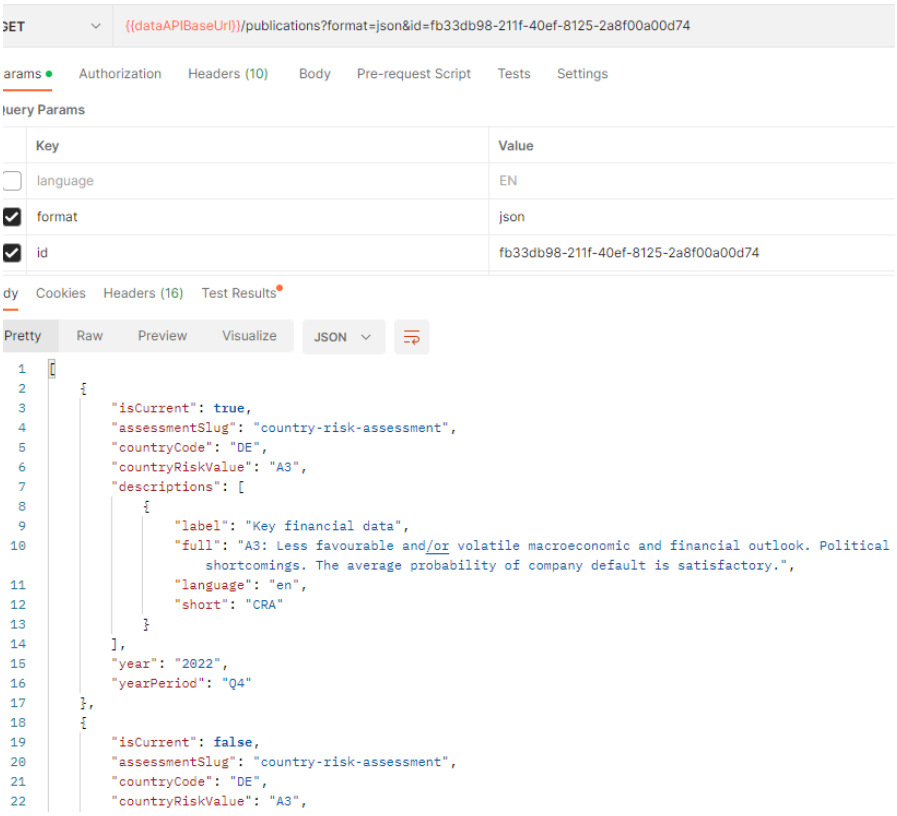


As both publications are ready, CRA and SRA, you can get now the content of the products, but with a call per each product.

Getting the sector risk assessments:



Getting the country risk assessment:



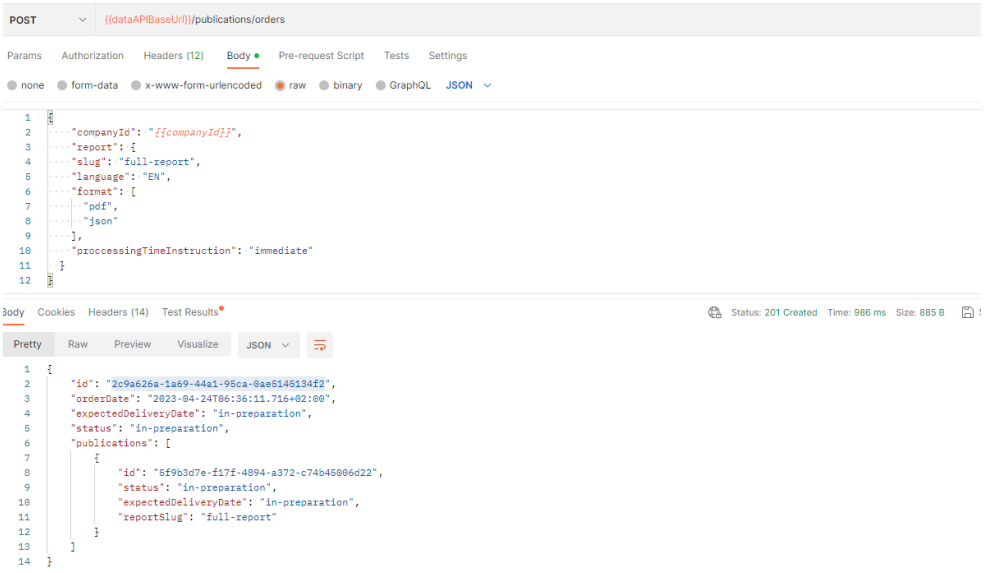
For ordering more products in one call, there is a request called URBA products inside postman collection on https://coface.github.io/DataAPI.html#documentation

2.1.2 Ordering other products for the same company

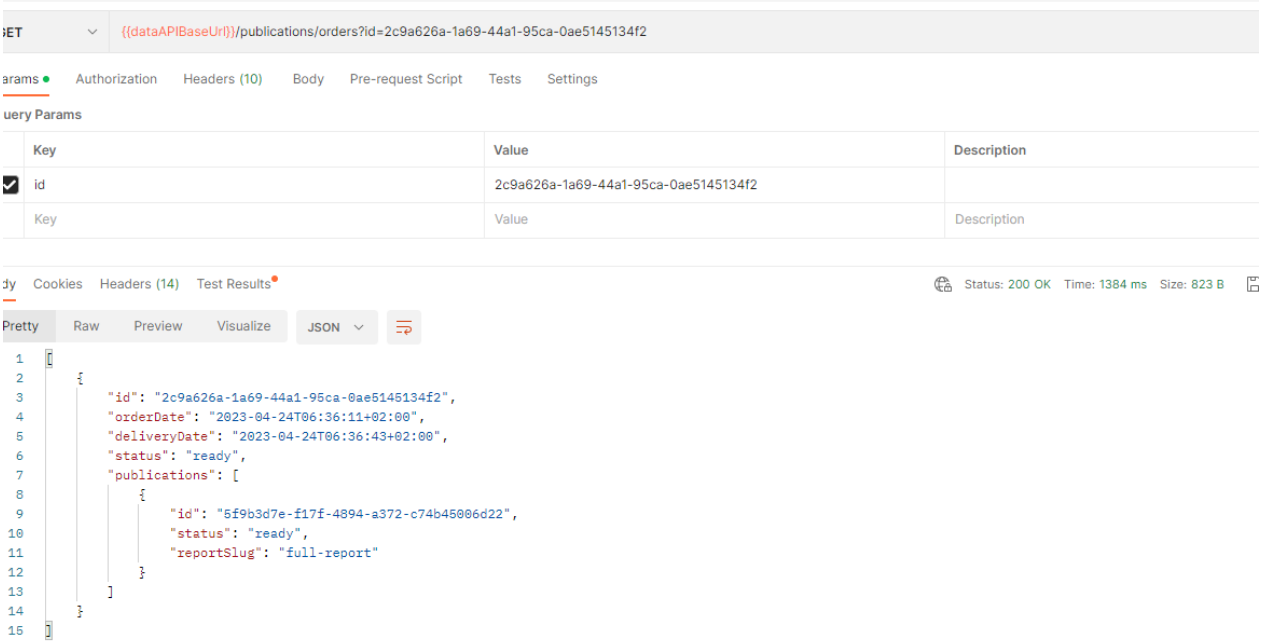
You can order now also reports.

We saw earlier when we checked the availability of the products that full report and snapshot report are immediate. Let's order them:

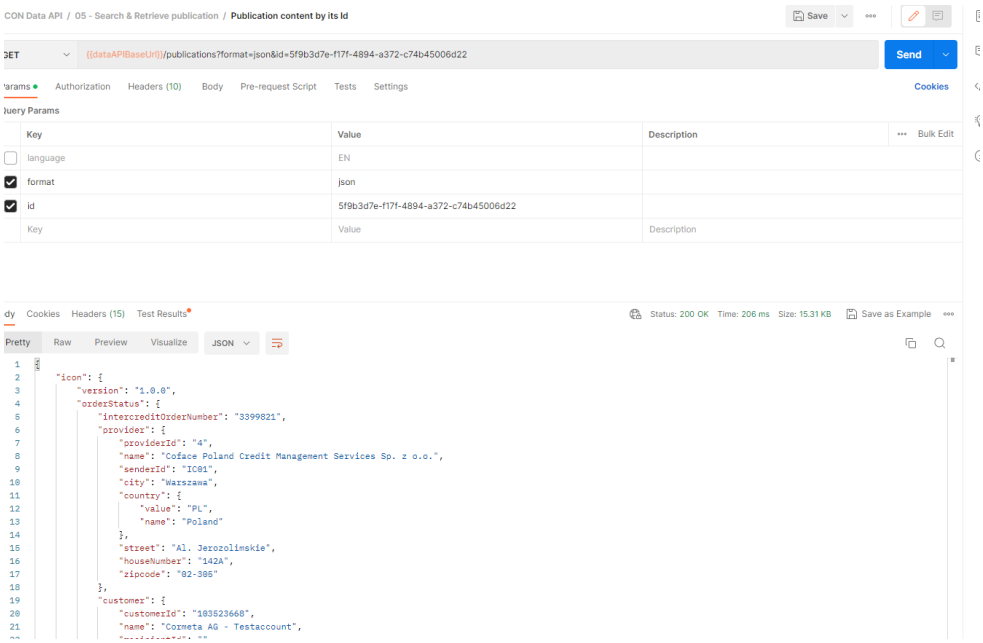
Ordering a full report:



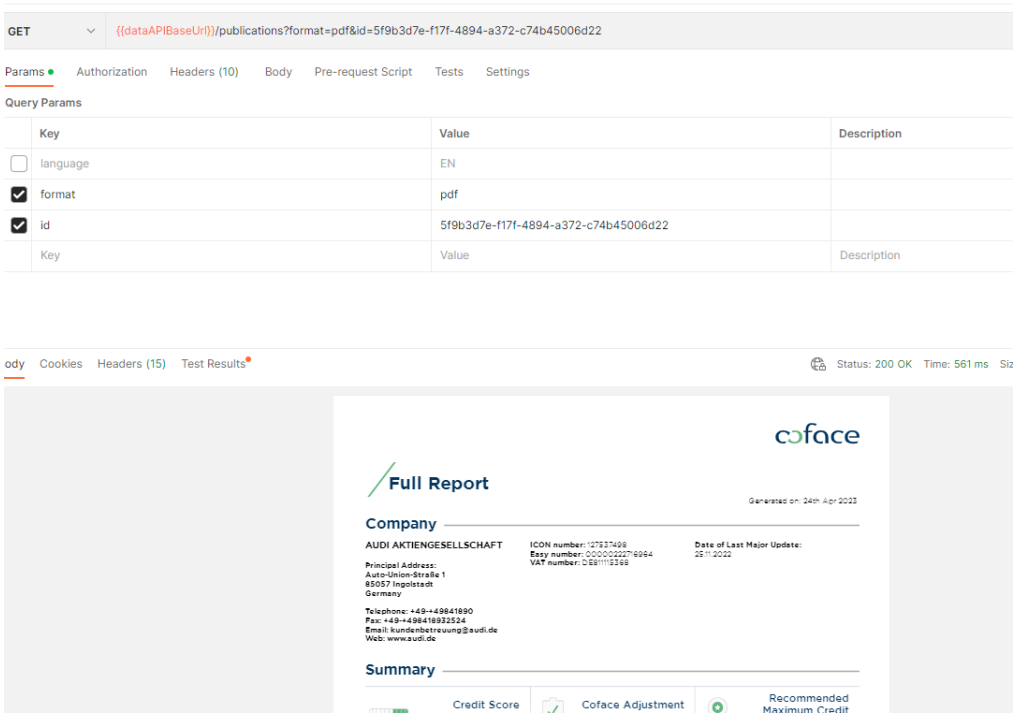
Checking the status of the order:



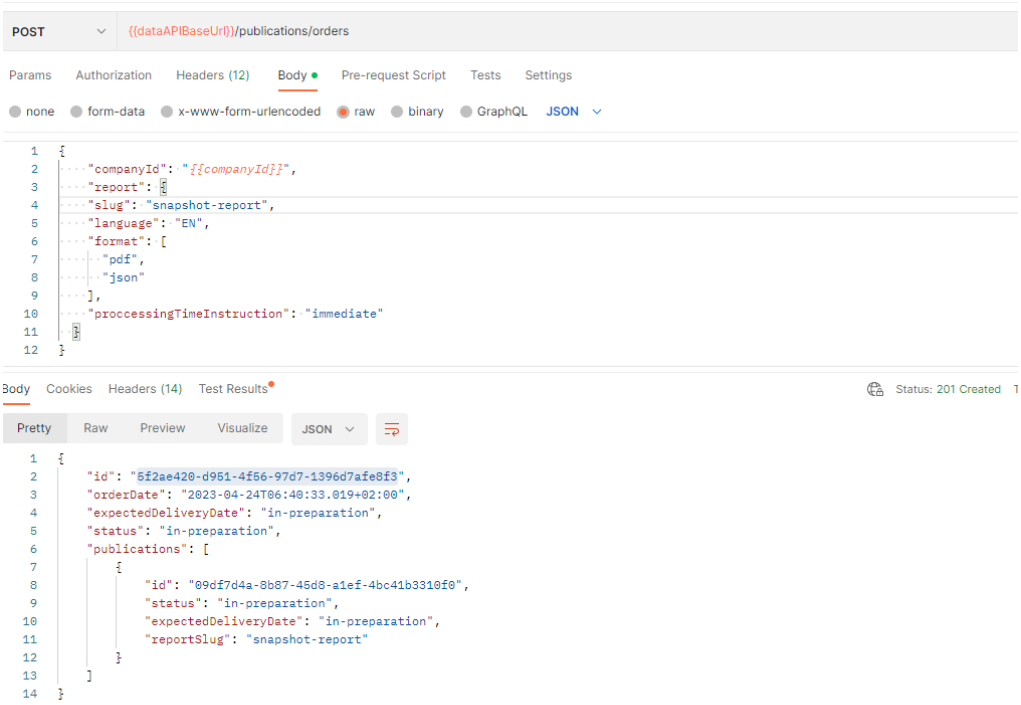
Getting the content of the full-report in json:



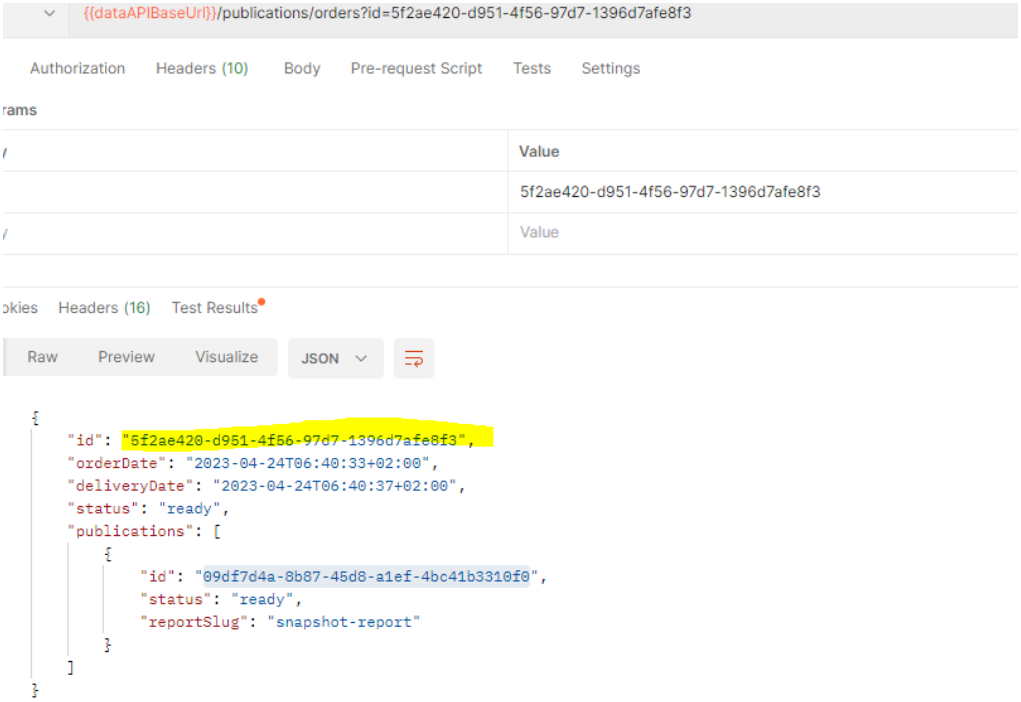
Getting the content of the full-report in pdf:



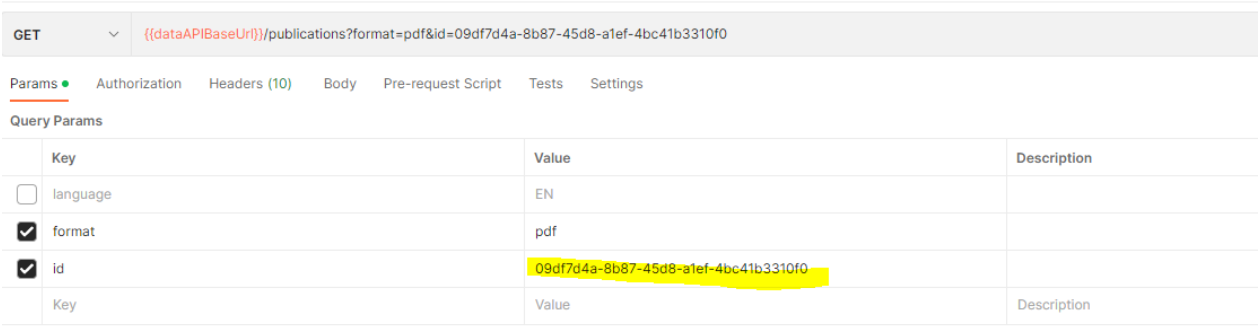
Now, let us order a snapshot report:



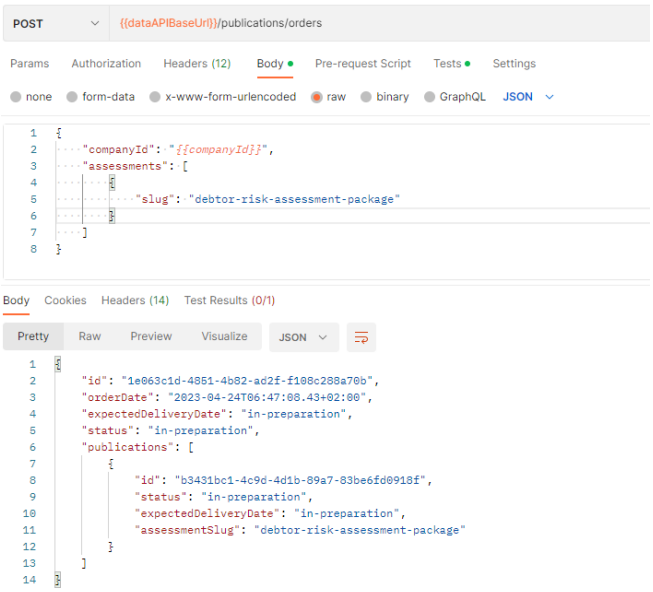
Getting order status:



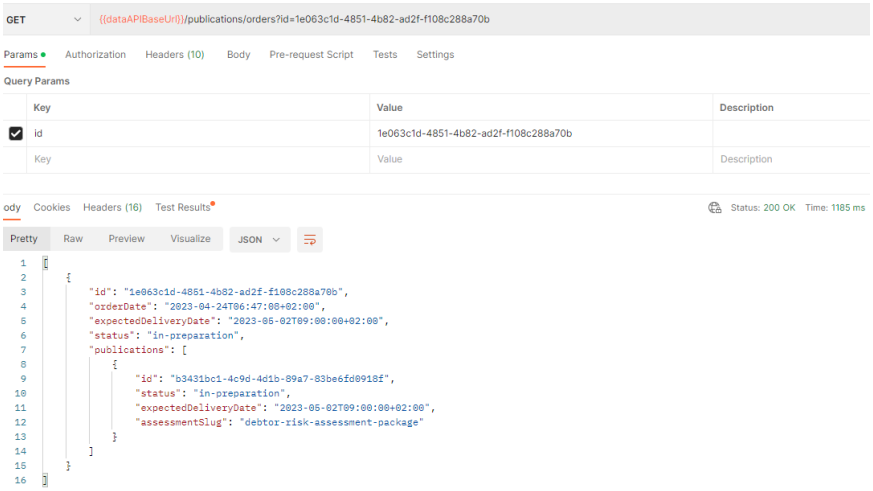
Getting the pdf format of the snapshot-report:



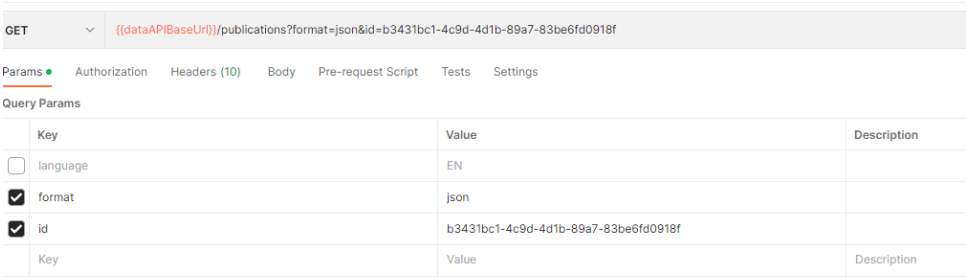
Let us now order one last product, Score product, DRA- package, which is the score with 5 years history:



As this is not an immediate product, and can take a little bit of time until is ready, first check status is still in-preparation:



If I am trying to get the content now, via /publications endpoint, it will not be ready:



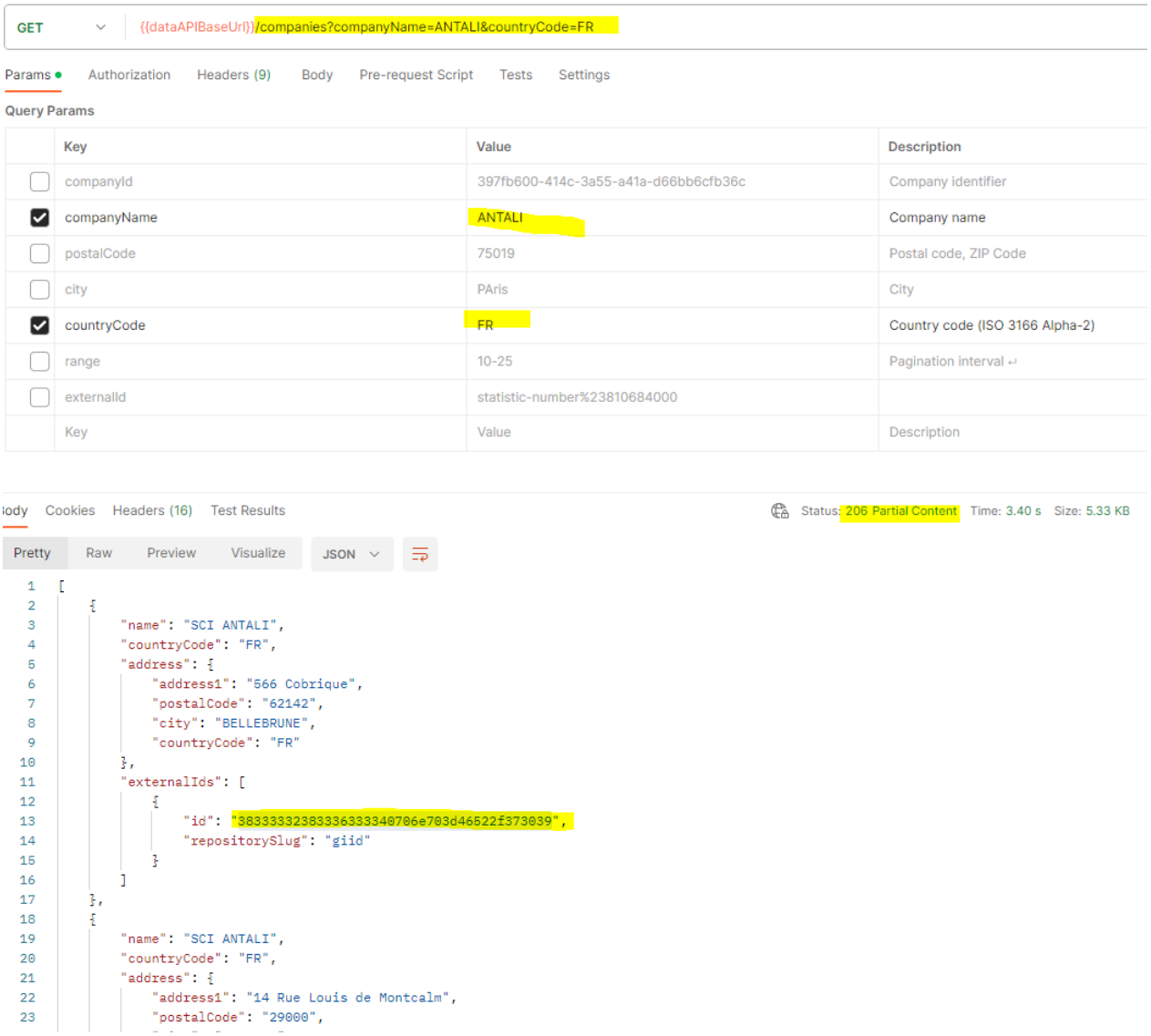
But we do have the expected delivery date which is the maximum SLA for the product to be available.

2.2 Ordering Late Payment Index product for a company

You want to order Late Payment Index for Antalis company in FR

First step is to search, using company name and country code, as this is what you know currently:

Call is sent to one of our providers (the one to which your customer profile is routed to). Provider is returning all results that it has with this name and country code. In this case, API returned 28 results.
Per page we have 20 results, and pagination is in place (you can see the 206 partial content, and headers values highlighted below):



GET {{dataAPIBaseUrl}}/companies?companyName=ANTALI&countryCode=FR

Params • Authorization Headers (9) Body Pre-request Script Tests Settings

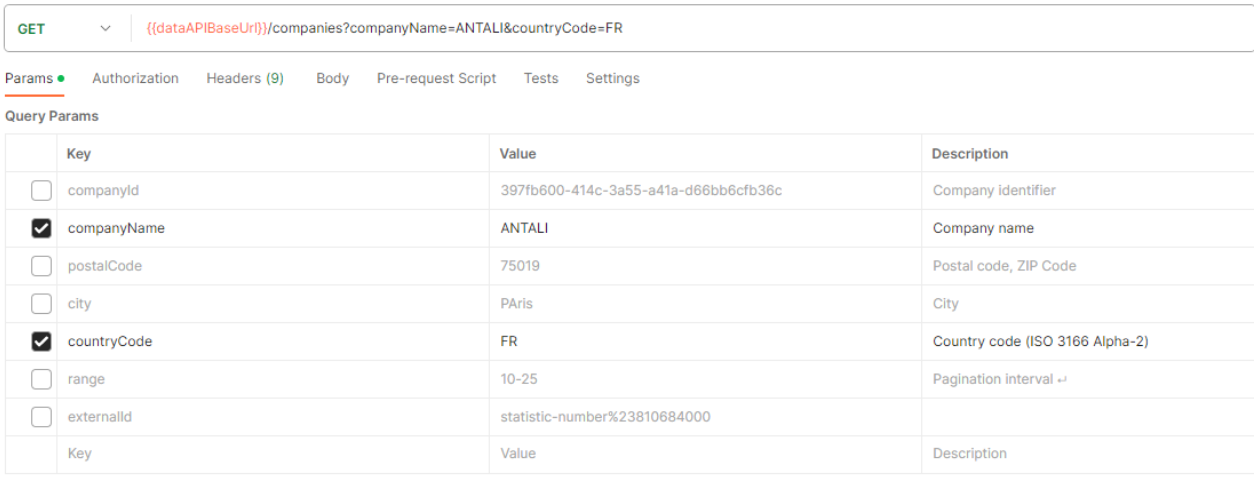
Query Params

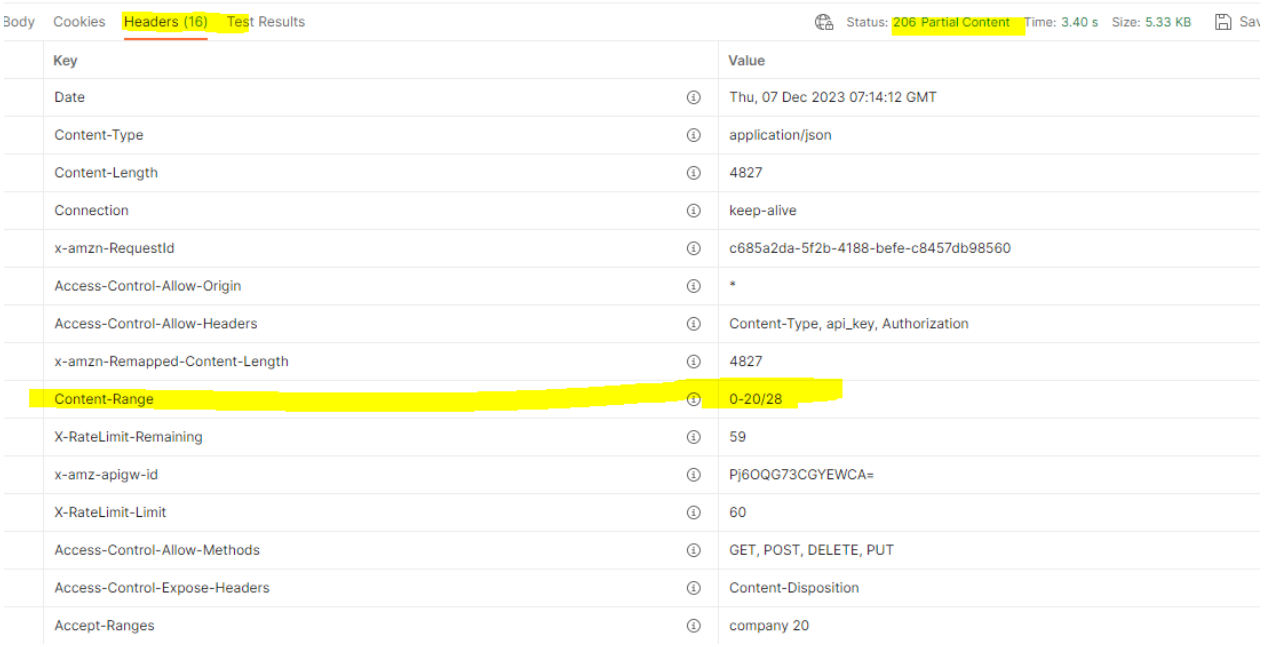
Key	Value	Description
companyId	397fb600-414c-3a55-a41a-d66bb6cfb36c	Company identifier
companyName	ANTALI	Company name
postalCode	75019	Postal code, ZIP Code
city	PAris	City
countryCode	FR	Country code (ISO 3166 Alpha-2)
range	10-25	Pagination interval
externalId	statistic-number%23810684000	
Key	Value	Description

Body Cookies Headers (16) Test Results Status: 206 Partial Content Time: 3.40 s Size: 5.33 KB

Pretty Raw Preview Visualize JSON


```
[
    {
        "name": "SCI ANTALI",
        "countryCode": "FR",
        "address": {
            "address1": "566 Cobrique",
            "postalCode": "62142",
            "city": "BELLEBRUNE",
            "countryCode": "FR"
        },
        "externalIds": [
            {
                "id": "383333323833336333340706e703d46522f373039",
                "repositorySlug": "giid"
            }
        ]
    },
    {
        "name": "SCI ANTALI",
        "countryCode": "FR",
        "address": {
            "address1": "14 Rue Louis de Montcalm",
            "postalCode": "29000",
```



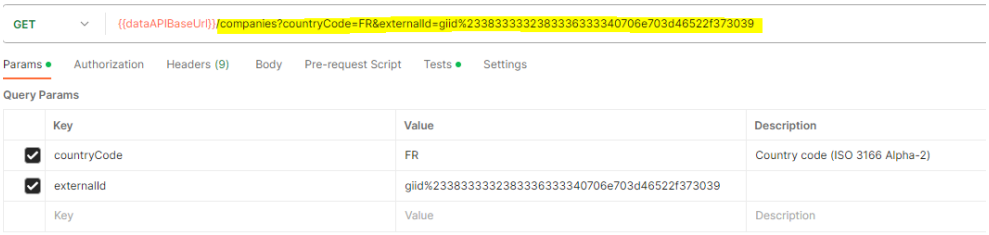


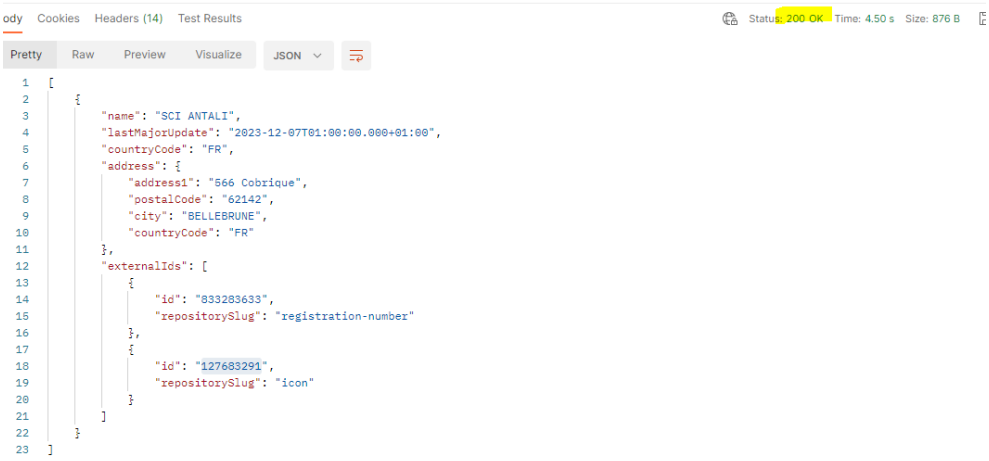
You have 28 results. In order to check others which are not in first page, you need to use the range parameter. We are showing max 20 results per page.

Now, out of 28 companies, you need to identify one of them in order to order some products.

There are 2 ways: either you choose one, and take the giid value and use it as externalId parameter and make the order with it, OR, you make one more call and uniquely identify the company and then you can order using another identifier replied by API (vat number, companyId, icon no)

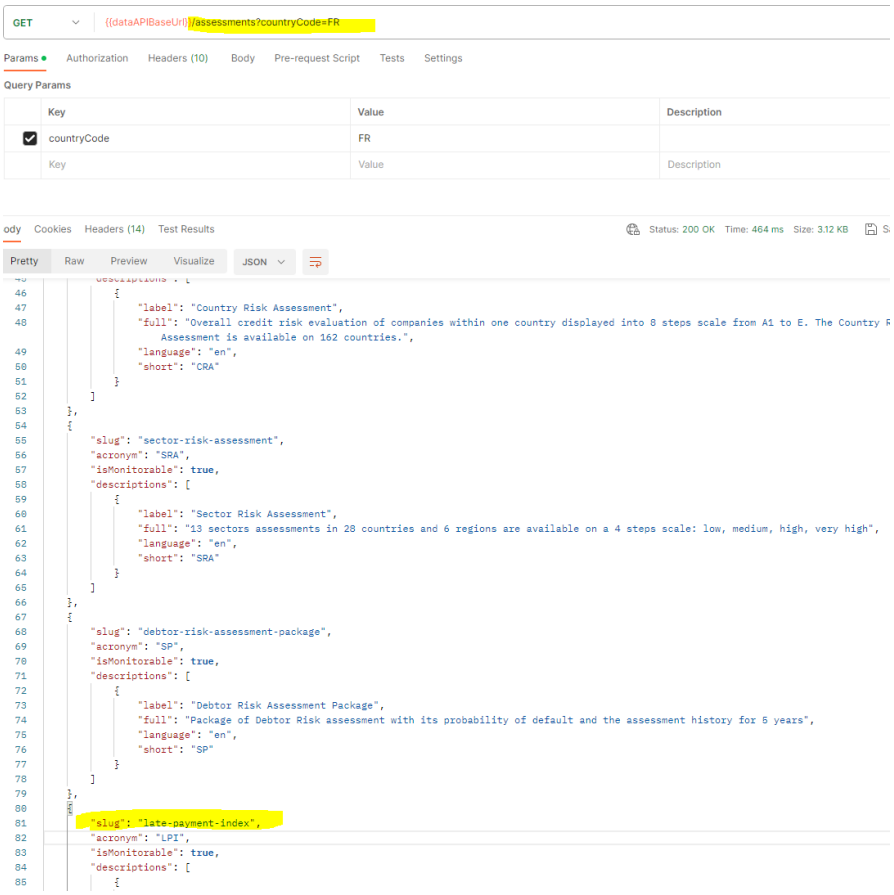
We are presenting below the second option:





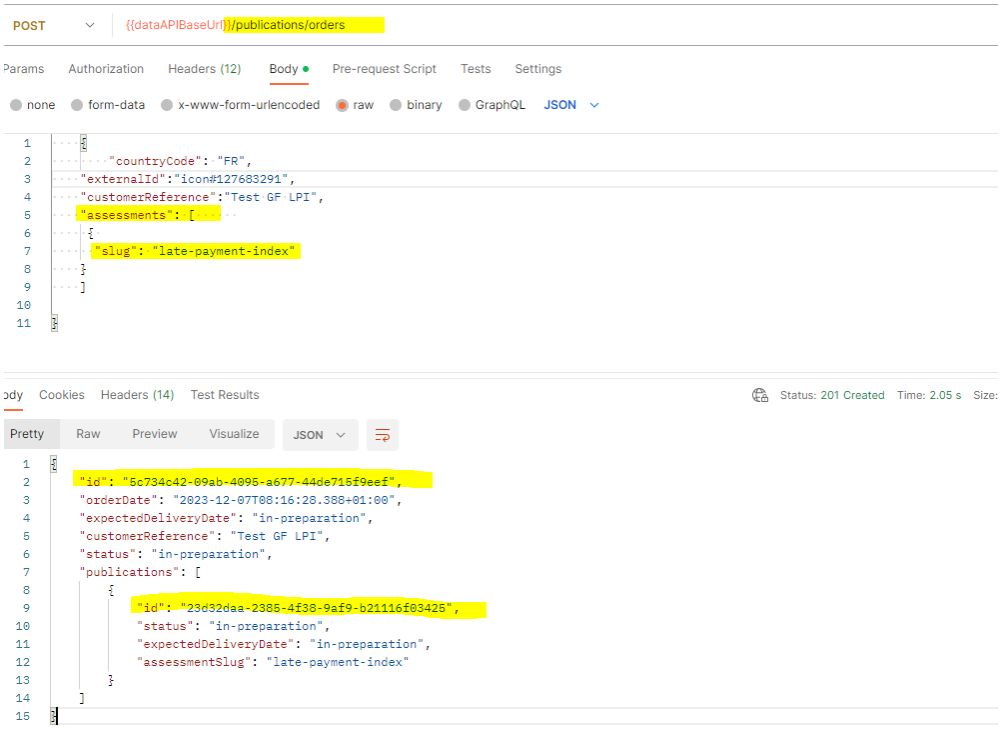
Now, company is saved into ICON, and has more identifiers associated: icon nr, registration-number.

In this moment, you need to check and see what are the available assessment products for this company, together with their availability (immediate or on-order). Late Payment Index should be returned:



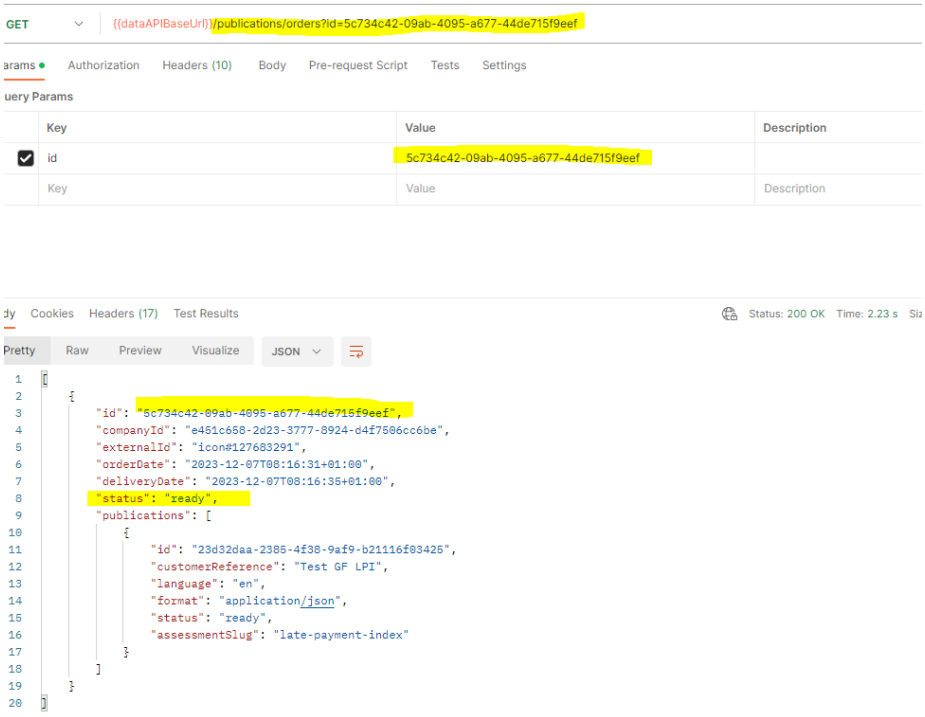
Now, you can order the product for this company:

You can use any of the well known options: via companyId (if exists), via other externalId: iconno, vat number, etc.



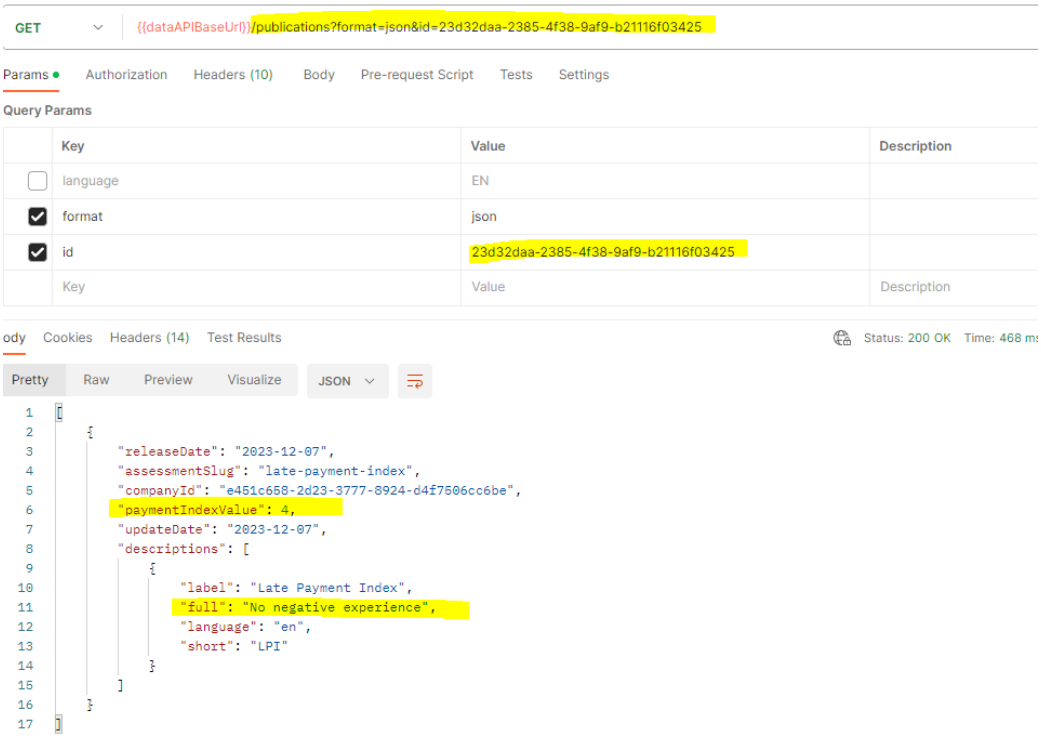
Ordering process is started. Fron now on, you are following the well known process: check the status of the order and then get the content.

Check the status of the order: GET /publications/orders?{orderId}



If the status is “ready”, you can get the content of the product:

GET /publications?format={format}&id={publicationId}


GET
{{dataAPIBaseUrl}}/publications?format=json&id=23d32daa-2385-4f38-9af9-b21116f03425
Params
Authorization
Headers (10)
Body
Pre-request Script
Tests
Settings
Query Params
Key
Value
Description
language
EN
format
json
id
23d32daa-2385-4f38-9af9-b21116f03425
Key
Value
Description
ody
Cookies
Headers (14)
Test Results
Status: 200 OK
Time: 468 ms
Pretty
Raw
Preview
Visualize
JSON
[
{
"releaseDate": "2023-12-07",
"assessmentSlug": "late-payment-index",
"companyId": "e451c658-2d23-3777-8924-d4f7506cc6be",
"paymentIndexValue": 4,
"updateDate": "2023-12-07",
"descriptions": [
{
"label": "Late Payment Index",
"full": "No negative experience",
"language": "en",
"short": "LPI"
}
]
}
]